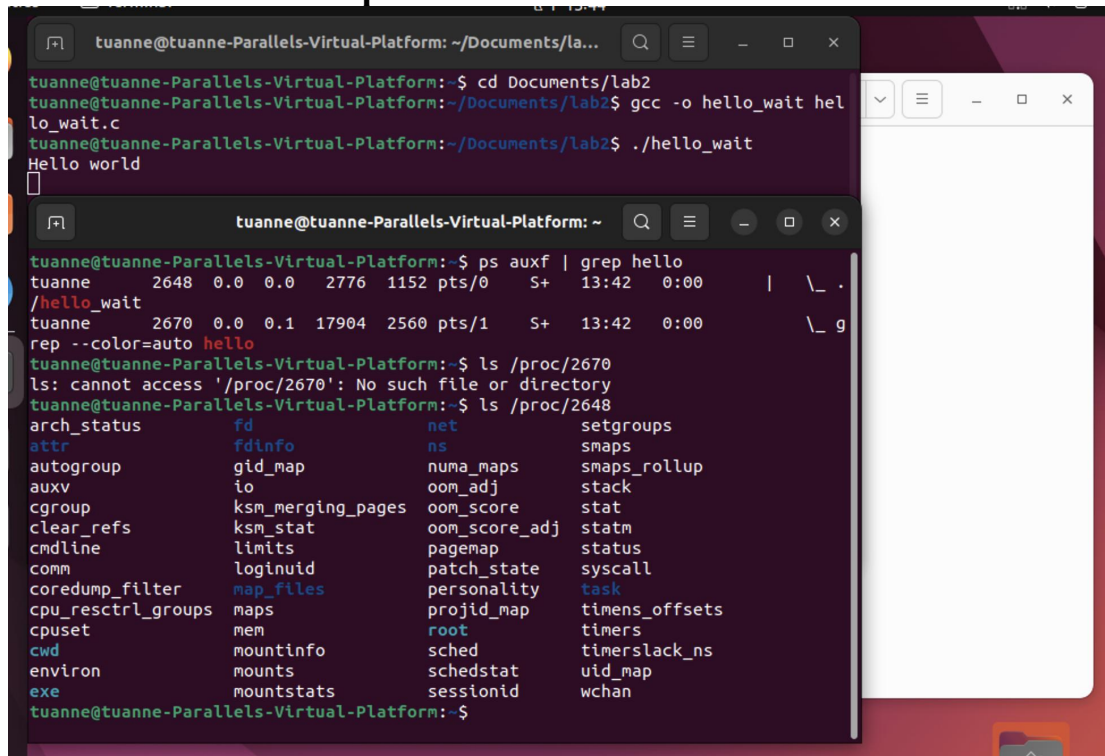


3.1 Practice 1: Create process



```
tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2
tuanne@tuanne-Parallels-Virtual-Platform:~/Documents/lab2$ gcc -o hello_wait hello_wait.c
tuanne@tuanne-Parallels-Virtual-Platform:~/Documents/lab2$ ./hello_wait
Hello world
^C

tuanne@tuanne-Parallels-Virtual-Platform: ~
tuanne@tuanne-Parallels-Virtual-Platform:~$ ps auxf | grep hello
tuanne    2648  0.0  0.0  2776  1152 pts/0    S+   13:42   0:00      | \_ .
./hello_wait
tuanne    2670  0.0  0.1  17904  2560 pts/1    S+   13:42   0:00      \_ g
rep --color=auto hello
tuanne@tuanne-Parallels-Virtual-Platform:~$ ls /proc/2670
ls: cannot access '/proc/2670': No such file or directory
tuanne@tuanne-Parallels-Virtual-Platform:~$ ls /proc/2648
arch_status      fd               net              setgroups
attr             fdinfo           ns               smaps
autogroup        gid_map          numa_maps       smaps_rollup
auxv             io              oom_adj         stack
cgroup          ksm_merging_pages oom_score       stat
clear_refs      ksm_stat         oom_score_adj  statm
cmdline         limits           pagemap        status
comm            loginuid         patch_state    syscall
coredump_filter map_files        personality     task
cpu_resctrl_groups maps             projid_map     timens_offsets
cpuset          mem             root           timers
cwd             mountinfo       sched          timerslack_ns
environ         mounts          schedstat      uid_map
exe             mountstats      sessionid      wchan
tuanne@tuanne-Parallels-Virtual-Platform:~$
```

- **ps auxf | grep hello** : để tìm các các process đang thực thi và lọc kết quả chỉ hiện thi liên quan tới hello
- **ls /proc/<pid>** : hiện thi nội dung của ứng với thư mục mang pid trong /proc

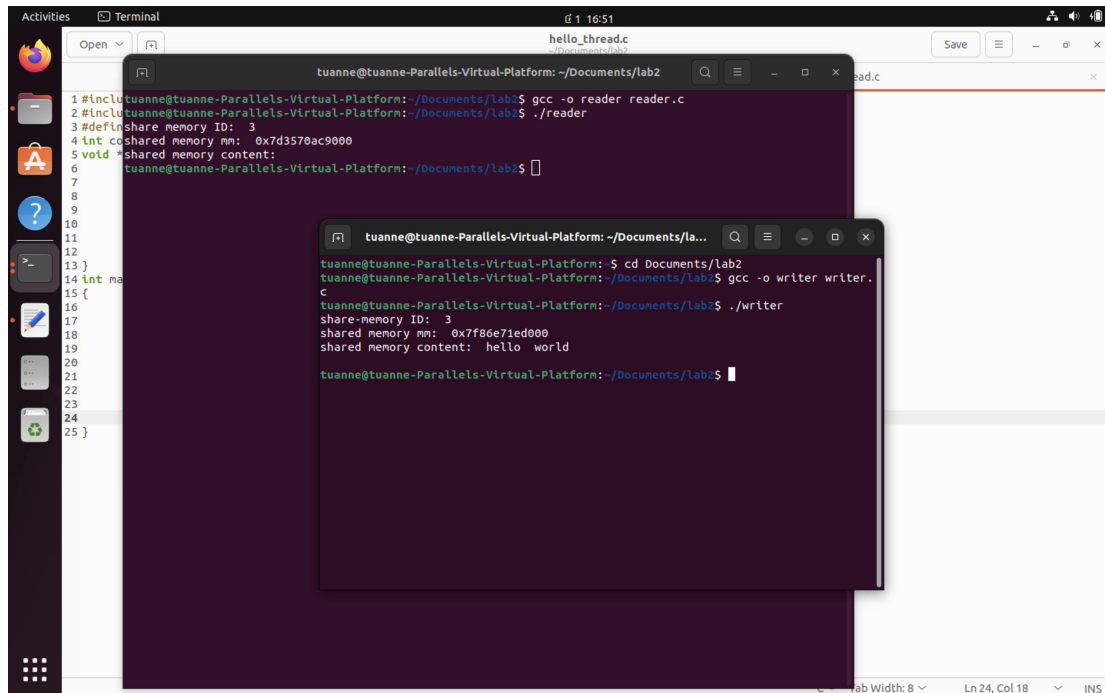
3.2 Practice 2: Traverse the tree of processes

-output đầu ra của các biến data và func là địa chỉ lưu trữ của chúng.

-cat /proc/<pid>/maps: để hiển thị nội dung của tệp "maps" trong thư mục /proc/<pid>

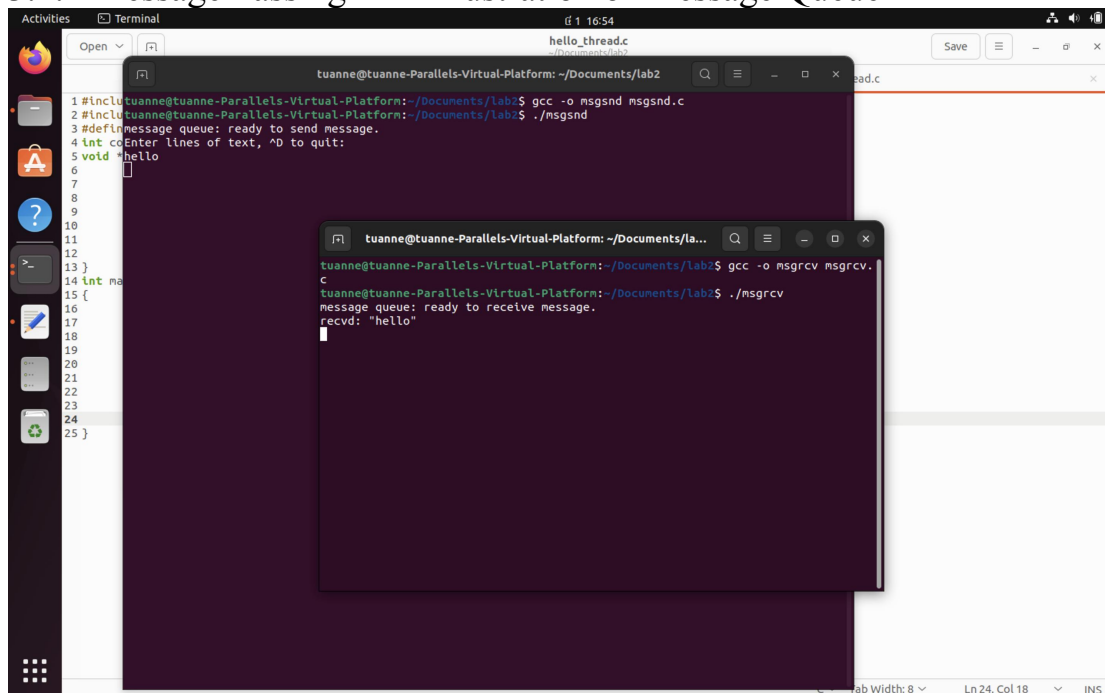
3.4 Practice 4: Inter Process Communication

3.4.1 Shared Memory



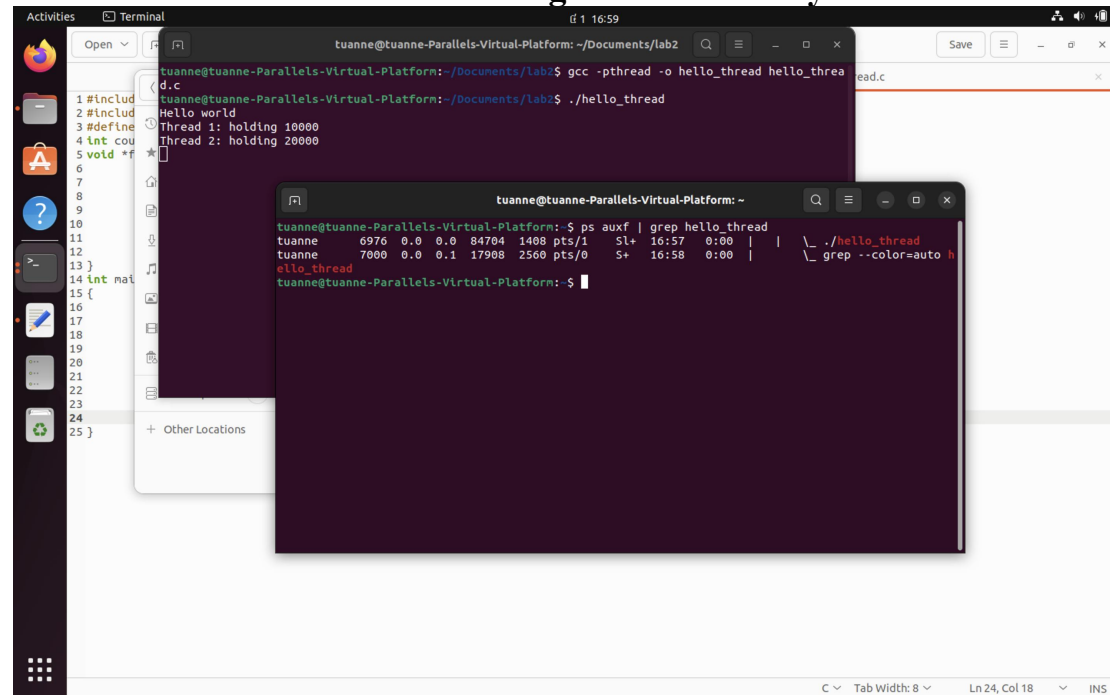
```
tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2
1 #include <stdio.h>
2 #include <unistd.h>
3 #define SHM_ID 3
4 int main() {
5     void *shm;
6     shm = mmap(0, 4096, PROT_READ | PROT_WRITE, MAP_SHARED, -1, SHM_ID);
7     printf("shared memory address: %p\n", shm);
8     printf("shared memory content: %s\n", (char *)shm);
9     return 0;
10 }
11
12 tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$ gcc -o reader reader.c
13 tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$ ./reader
14 shared memory address: 0x7d3570ac9000
15 shared memory content:
16
17 tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$
18
19 tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$ cd Documents/lab2
20 tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$ gcc -o writer writer.c
21 tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$ ./writer
22 shared memory address: 0x7d3570ac9000
23 shared memory content: hello world
24
25 tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$
```

3.4.2 Message Passing - An illustration of Message Queue



```
tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2
1 #include <stdio.h>
2 #include <unistd.h>
3 #include <sys/types.h>
4 #include <sys/ipc.h>
5 #include <sys/msg.h>
6 #define MSG_ID 3
7 int main() {
8     struct msgbuf {
9         long mtype;
10        char mtext[1024];
11    };
12    struct msgbuf msg;
13    msg.mtype = MSG_ID;
14    printf("Enter lines of text, ^D to quit:\n");
15    while (1) {
16        fgets(msg.mtext, 1024, stdin);
17        if (!feof(stdin)) {
18            msgsnd(IPC_PRIVATE, &msg, 1, 0);
19            printf("Message sent.\n");
20        }
21    }
22    return 0;
23 }
24
25 tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$ gcc -o msgsnd msgsnd.c
26 tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$ ./msgsnd
27 Enter lines of text, ^D to quit:
28 hello
29
30 tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$
31
32 tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$ cd Documents/lab2
33 tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$ gcc -o msgrcv msgrcv.c
34 tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$ ./msgrcv
35 message queue: ready to receive message.
36 recvd: "hello"
37
38 tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$
```

3.5 Practice 5: Create thread using Pthread library



```
1 #include <stdio.h>
2 #include <pthread.h>
3 #define HELLO_WORLD "Hello world"
4 int count = 0;
5 void *thread_func(void *arg)
6 {
7     printf("Thread 1: holding 10000\n");
8     sleep(10);
9     count++;
10 }
11
12 int main()
13 {
14     pthread_t thread;
15     pthread_create(&thread, NULL, thread_func, NULL);
16     printf("Hello world\n");
17     sleep(10);
18     pthread_join(thread, NULL);
19     printf("Thread 2: holding 20000\n");
20     return 0;
21 }
22
23
24
25 }
```

```
tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$ gcc -pthread -o hello_thread hello_thread.c
tuanne@tuanne-Parallels-Virtual-Platform: ~/Documents/lab2$ ./hello_thread
Hello world
Thread 1: holding 10000
Thread 2: holding 20000

tuanne@tuanne-Parallels-Virtual-Platform: ~$ ps auxf | grep hello_thread
tuanne    6976  0.0  0.0 84704 1408 pts/1    Sl+  16:57   0:00 |  \_ ./hello_thread
tuanne    7000  0.0  0.1 17908 2560 pts/0    S+   16:58   0:00 |  \_ grep --color=auto h
hello_thread
tuanne@tuanne-Parallels-Virtual-Platform: ~$
```