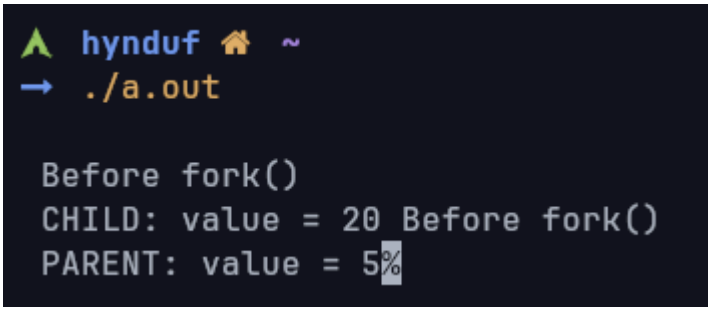
BÀI LÀM PROGRAMMING-EX-2

Huỳnh Tiến Dũng - 21020007

1. Chạy chương trình ở Slide 17 - NLHĐH-Bai-2

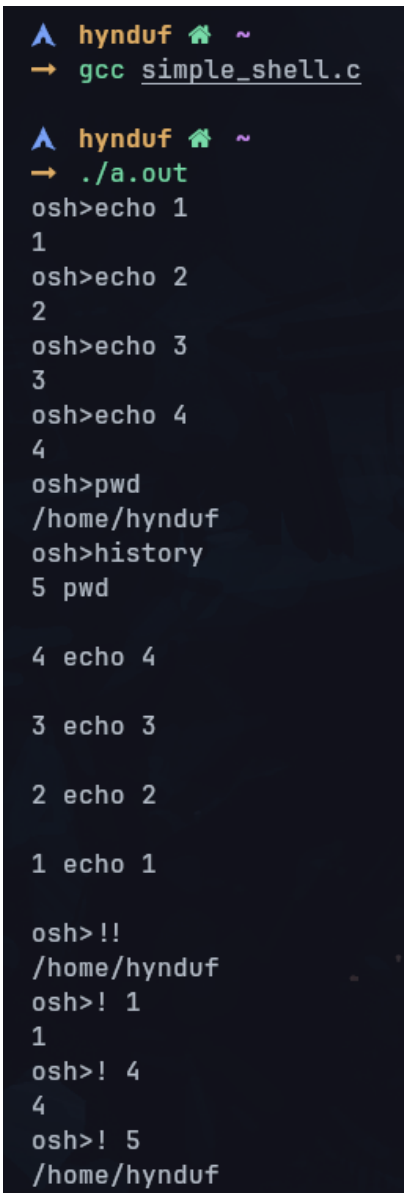
Mã nguồn nằm trong file myfork.c
Output:



Lí do “Before fork()” bị lặp lại là do chưa flush output ra khỏi buffer. Nếu thêm fflush(stdout); hoặc thêm \n vào cuối Before fork() trong printf thì sẽ chỉ còn một “Before fork()” trong output.

2. Chapter 3 - Project 1

Mã nguồn nằm trong file chapter3-project1.c



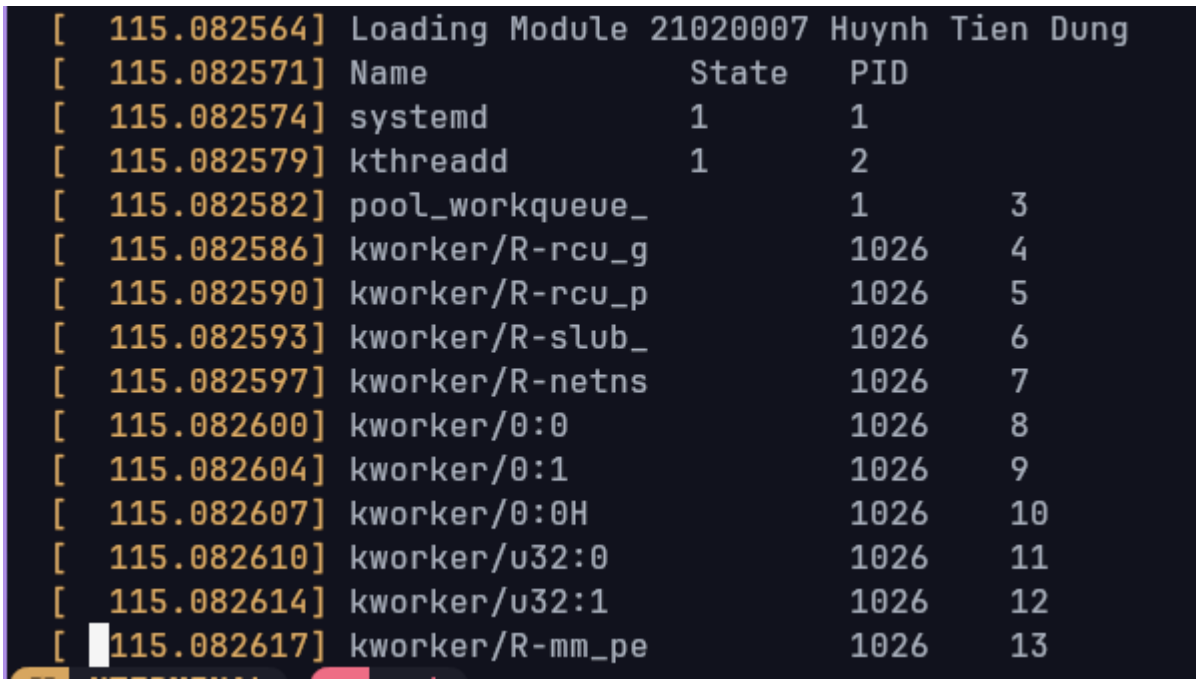
3. Chapter 3 - Project 2 - Part 1

Mã nguồn nằm trong file chapter3-project2-part1.c
Phần chính:

```
for_each_process(task)
{
        printk(KERN_INFO "%s\t\t%ld\t%d\n", task->comm, (long)task->__state, task->pid);
}
```

task->comm là name của process
task->__state là trạng thái của process
task->pid là pid của process

Output dmesg (vài chỗ bị lệch do tên của các process dài hơn 1 \t):

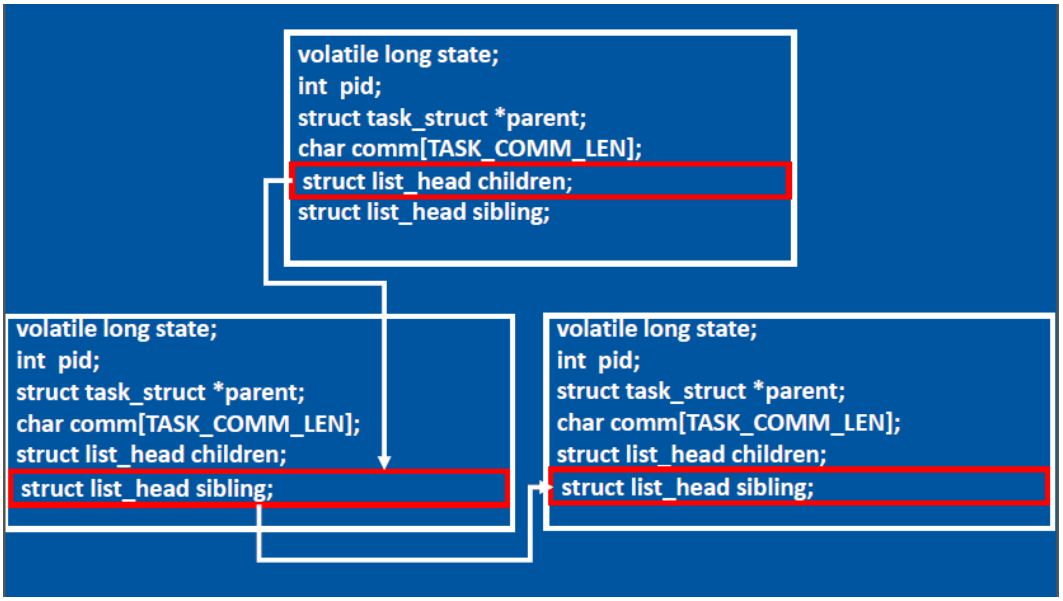


4. Chapter 3 - Project 2 - Part 2

Mã nguồn nằm trong file chapter3-project2-part2.c
Phần chính:

```
void depth_first_search_tasks(struct list_head *list, struct list_head *next)
{
    list_for_each(list, next)
    {
        task = list_entry(list, struct task_struct, sibling);
        printk(KERN_INFO "%s\t\t\t%ld\t%d\n", task->comm, (long) task->__state,
task->pid);
        depth_first_search_tasks(list, &task->children);
    }
}
```

Cấu trúc con trỏ children và sibling trong 1 task_struct:



Đây giải thích cho dòng code:
task = list_entry(list, struct task_struct, sibling);
(list_entry() sẽ lấy ra được task_struct mà có biến ‘sibling' chính là con trỏ ‘list', ‘list' ban đầu khi for sẽ từ children để trỏ tới sibling đầu tiên ở dưới và lần lượt đi hết các siblings)

Output dmesg (vài chỗ bị lệch do tên của các process dài hơn 1 \t):

```
[ 1396.244064] Loading Module 21020007 Huynh Tien Dung
[ 1396.244067] Name                    State   PID
[ 1396.244068] systemd                 1       1
[ 1396.244070] systemd-journal                 1       306
[ 1396.244071] systemd-udevd                   1       346
[ 1396.244073] bluetoothd                      1       538
[ 1396.244074] dbus-daemon                     1       539
[ 1396.244075] dhcpcd                  1       540
[ 1396.244076] dhcpcd                  1       542
[ 1396.244077] dhcpcd                  1       687
[ 1396.244079] dhcpcd                  1       543
[ 1396.244080] dhcpcd                  1       544
[ 1396.244081] systemd-logind                  1       545
```

5. Bài tập 3.12

i = 0. Sau fork sẽ có 2 process (1 cha, 1 con)
i = 1. Sau fork sẽ có 4 process
i = 2. Sau fork sẽ có 8 process
i = 3. Sau fork sẽ có 16 process

Vậy tổng cộng có 16 process

6. Bài tập 3.14

A: pid = 0
B: pid = 2603
C: pid = 2603
D: pid = 2600