

```

#include <linux/module.h>
#include <linux/kernel.h>
// part 2
#include <linux/sched.h>
// part 2 extra
#include <linux/hash.h>
#include <linux/gcd.h>
#include <asm/param.h>
#include <linux/jiffies.h>

void print_init_PCB(void)
{
    printk(KERN_INFO "init_task pid:%d\n", init_task.pid);
    printk(KERN_INFO "init_task state:%lu\n", init_task.state);
    printk(KERN_INFO "init_task flags:%d\n", init_task.flags);
    printk(KERN_INFO "init_task runtime priority:%d\n", init_task.rt_priority);
    printk(KERN_INFO "init_task process policy:%d\n", init_task.policy);
    printk(KERN_INFO "init_task task group id:%d\n", init_task.tgid);
}

/* This function is called when the module is loaded. */
int simple_init(void)
{
    printk(KERN_INFO "Loading Module\n");

    print_init_PCB();
    printk(KERN_INFO "Golden Ration Prime = %lu\n", GOLDEN_RATIO_PRIME);
    printk(KERN_INFO "HZ = %d\n", HZ);
    printk(KERN_INFO "enter jiffies = %lu\n", jiffies);

    return 0;
}

/* This function is called when the module is removed. */
void simple_exit(void) {
    printk(KERN_INFO "Removing Module\n");
    printk(KERN_INFO "gcd of 3300 and 24 = %lu\n", gcd(3300,24));
    printk(KERN_INFO "exit jiffies = %lu\n", jiffies);
}

/* Macros for registering module entry and exit points. */
module_init( simple_init );
module_exit( simple_exit );

MODULE_LICENSE("GPL");
MODULE_DESCRIPTION("Simple Module");
MODULE_AUTHOR("SGG");

```