DRVO

H2 Character device driver

H2.2

dit is mijn .c file die ik gebruikt hebt

```
#include <linux/init.h>
#include <linux/module.h>
MODULE_LICENSE("Dual BSD/GPL");
static int hello_init(void)
{
    printk(KERN_ALERT "Hello, world\n");
    return 0;
}
static void hello_exit(void)
{
    printk(KERN_ALERT "Goodbye, world\n");
}
module_init(hello_init);
module_exit(hello_exit);
```

H2.3

waneer ik dit commando run met de meegegeven Makefile dan krijg ik het resultaat.

```
make hello.ko obj-m=hello.o -C /lib/modules/4.15.0-54-generic/build M=/home/joan/drvo/
modules
```

resultaat:

```
niels@niels-virtual-machine:~/github/DRVO/src$ make
make -C /lib/modules/6.5.0-28-generic/build M=/home/niels/github/DRVO/src modules
make[1]: Entering directory '/usr/src/linux-headers-6.5.0-28-generic'
warning: the compiler differs from the one used to build the kernel
  The kernel was built by: x86_64-linux-gnu-gcc-12 (Ubuntu 12.3.0-1ubuntu1~22.04)
12.3.0
  You are using:           gcc-12 (Ubuntu 12.3.0-1ubuntu1~22.04) 12.3.0
  CC [M]  /home/niels/github/DRVO/src/hello.o
  MODPOST /home/niels/github/DRVO/src/Module.symvers
  CC [M]  /home/niels/github/DRVO/src/hello.mod.o
  LD [M]  /home/niels/github/DRVO/src/hello.ko
  BTF [M] /home/niels/github/DRVO/src/hello.ko
```

```
Skipping BTF generation for /home/niels/github/DRVO/src/hello.ko due to unavailability
of vmlinux
make[1]: Leaving directory '/usr/src/linux-headers-6.5.0-28-generic'
```

ik krijg wat warnings maar kan het bouwen en testen.

H2.4

Als eerste heb ik in een anderesource,.shell de commando:

```
sudo dmesg -w
```

Om ervoor te zorgen dat ik printk kan meekijken.

Dan doe ik het commando:

```
sudo insmod hello.ko
```

En krijg ik het resultaat in de log te zien:

```
[ 5280.572849] Hello, world
```

En wanneer ik het commando:

```
sudo rmmod hello
```

krijg het resultaat te zien in de log:

```
[ 5362.805165] Goodbye, world
```

3 Bouw een standaard device driver

H3.1

source code:

```
#include <linux/init.h>
#include <linux/module.h>
MODULE_LICENSE("Dual BSD/GPL");
static int hello_init(void)
{
    printk(KERN_ALERT "Hello, world\n");
```

```
    return 0;
}
static void hello_exit(void)
{
    printk(KERN_ALERT "Goodbye, world\n");
}
module_init(hello_init);
module_exit(hello_exit);
```

Commando om te bouwen:

```
make
```

Kernel resultaat bij registeren en de-registreren:

```
[ 6537.263969] Hello, world
[ 6539.340526] Goodbye, world
```

Dat de module in /proc/modules staat als de module geregistreerd is:

```
niels@niels-virtual-machine:~/github/DRVO$ cat /proc/modules
hello 12288 0 - Live 0x0000000000000000 (OE)
tls 151552 0 - Live 0x0000000000000000
isofs 61440 2 - Live 0x0000000000000000
vsock_loopback 12288 0 - Live 0x0000000000000000
vmw_vsock_virtio_transport_common 57344 1 vsock_loopback, Live 0x0000000000000000
intel_rapl_msr 20480 0 - Live 0x0000000000000000
vmw_vsock_vmci_transport 45056 1 - Live 0x0000000000000000
vsock 61440 5
vsock_loopback,vmw_vsock_virtio_transport_common,vmw_vsock_vmci_transport, Live
0x0000000000000000
intel_rapl_common 40960 1 intel_rapl_msr, Live 0x0000000000000000
vmw_balloon 28672 0 - Live 0x0000000000000000
```

Dat de module staat in lsmod als de module geregistreerd is:

```
niels@niels-virtual-machine:~/github/DRVO/src$ sudo lsmod
Module                  Size  Used by
hello                  12288  0
tls                   151552  0
isofs                  61440  2
```

H3.2

```
:language: Makefile

inputFile=opgave_3_3
```

```
KDIR := /lib/modules/$(shell uname -r)/build
PWD := $(shell pwd)

obj-m = $(inputFile).o

all:
    $(MAKE) -C $(KDIR) M=$(PWD) modules

%.ko:%.c
    $(MAKE) -C $(KDIR) M=$(PWD) modules

clean:
    $(MAKE) -C $(KDIR) M=$(PWD) clean
    rm -f *.o *.ko *.order *.cmd *.symvers *.mod.c
    rm -rf .tmp_versions
```

H3.3

```
make -C /lib/modules/6.5.0-28-generic/build M=/home/niels/github/DRVO/src modules
make[1]: Entering directory '/usr/src/linux-headers-6.5.0-28-generic'
warning: the compiler differs from the one used to build the kernel
  The kernel was built by: x86_64-linux-gnu-gcc-12 (Ubuntu 12.3.0-1ubuntu1~22.04)
12.3.0
  You are using:           gcc-12 (Ubuntu 12.3.0-1ubuntu1~22.04) 12.3.0
  CC [M]  /home/niels/github/DRVO/src/opgave_3_3.o
  MODPOST /home/niels/github/DRVO/src/Module.symvers
  CC [M]  /home/niels/github/DRVO/src/opgave_3_3.mod.o
  LD [M]  /home/niels/github/DRVO/src/opgave_3_3.ko
  BTF [M] /home/niels/github/DRVO/src/opgave_3_3.ko
Skipping BTF generation for /home/niels/github/DRVO/src/opgave_3_3.ko due to
unavailability of vmlinux
make[1]: Leaving directory '/usr/src/linux-headers-6.5.0-28-generic'
```

H3.4 en H3.5

```
#include <linux/init.h>
#include <linux/module.h>
#include <linux/cdev.h>
#include <linux/fs.h>
MODULE_LICENSE("Dual BSD/GPL");


static const int major = 500;
static const int minor = 0;
static const int amount = 1; // amount of major nrs.
static const char driver_name[] = "hello_driver";
```

```
/* device structures */
static struct cdev* device;

/**
* Open/release
**/
static int hello_open(struct inode *inode, struct file *file)
{
    printk(KERN_ALERT "hello_open()\n");
    return 0;
}
static int hello_release(struct inode *inode, struct file *file)
{
    printk(KERN_ALERT "hello_release()\n");
    return 0;
}

/**
* read / write
**/
static ssize_t
hello_read(struct file *file, char __user * buf, size_t lbuf, loff_t * ppos)
{
    printk(KERN_ALERT "hello_read()\n");
    return 0;
}
static ssize_t
hello_write(struct file *file, const char __user * buf, size_t lbuf, loff_t * ppos)
{
    printk(KERN_ALERT "hello_write())\n");
    return lbuf;
}

struct file_operations fops = {
    .read = hello_read,
    .write = hello_write,
    .open = hello_open,
    .release = hello_release,
};

static int hello_init(void)
{
    dev_t device_number;
    int result;
    device_number = MKDEV(major, minor);
    device = cdev_alloc();
    if(!device){
        printk(KERN_ALERT "Failed ALLOCATION");
        return -ENOMEM;
    }
    cdev_init(device, &fops);
```

```
    result = register_chrdev_region(device_number, amount, driver_name);
    if (result < 0) {
        printk(KERN_ALERT "Failed to register device region: %d\n", result);
        return result;
    }

        result = cdev_add(device, device_number, amount);
        if (result < 0) {
            printk(KERN_ALERT "Failed to add cdev: %d\n", result);
            unregister_chrdev_region(device_number, amount);
        return result;
        }

    printk(KERN_ALERT "hello_init()\n");
    return 0;
}
static void hello_exit(void)
{
    dev_t device_number;

    device_number = MKDEV(major, minor);
    cdev_del(device);

    unregister_chrdev_region(device_number, amount);

    printk(KERN_ALERT "hello_exit()\n");

}

module_init(hello_init);
module_exit(hello_exit);
```

```
sudo mknod /dev/hello_driver c 500 0 -m 0666
```

```
brw-rw----  1 root  disk      2,   0 apr 23 17:06 fd0
crw-rw-rw-  1 root  root      1,   7 apr 23 17:06 full
crw-rw-rw-  1 root  root     10, 229 apr 23 17:06 fuse
**crw-rw-rw-  1 root  root    500,   0 apr 23 19:11 hello_driver**
crw-------  1 root  root    241,   0 apr 23 17:06 hidraw0
crw-------  1 root  root     10, 228 apr 23 17:06 hpet
drwxr-xr-x  2 root  root           0 apr 23 17:00 hugepages
```

H3.5

H3.6

H3.7

H3.8