Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут ім. Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

**Лабораторна робота №6**

З дисципліни «Архітектура компʼютерів 2»

ВИКОНАВ:

студент ІІІ курсу ФІОТ

Групи ІО-82

Шендріков Є.О.

ПЕРЕВІРИВ:

Каплунов А. В.

Київ 2020 р.

**hello1.h**

#include <linux/ktime.h>

#pragma once

struct head\_list {

struct head\_list \*next;

ktime\_t start\_time;

ktime\_t finish\_time;

};

int hello\_print(uint count);

**hello1.c**

#include <linux/init.h>

#include <linux/module.h>

#include <linux/moduleparam.h>

#include <linux/printk.h>

#include <linux/slab.h>

#include <hello1.h>

MODULE\_AUTHOR("Jack Shendrikov <jackshen@ukr.net>");

MODULE\_DESCRIPTION("Hello, world in Linux Kernel Training");

MODULE\_LICENSE("Dual BSD/GPL");

MODULE\_VERSION("2.0");

static struct head\_list \*head = NULL;

int free\_list(int print\_smth) {

struct head\_list \*tmp;

if (ZERO\_OR\_NULL\_PTR(head)) return 0;

while (head != NULL) {

tmp = head;

if (print\_smth) pr\_info("Printing time: %lld", tmp->finish\_time - tmp->start\_time);

head = tmp->next;

kfree(tmp);

}

if (!ZERO\_OR\_NULL\_PTR(head)) return -EAGAIN;

pr\_info("Module removed!");

pr\_info("");

return 0;

}

int hello\_print(uint count) {

uint i = 0;

struct head\_list \*\*tmp1, \*\*tmp2;

if (!ZERO\_OR\_NULL\_PTR(head))

if (free\_list(0) != 0)

return -EAGAIN;

tmp1 = &head;

if (!count) {

pr\_warning("The parameter is 0. You cannot enter 0.");

}

if (count >= 5 && count <= 10) {

pr\_warning("The parameter is %d, between [5;10], enter number less than 5", count);

} else if (count > 10) {

pr\_err("The parameter is %d, (> 10)", count);

return -EINVAL;

}

for (i = 0; i < count; i++) {

\*tmp1 = kmalloc(sizeof(struct head\_list), GFP\_KERNEL);

if (ZERO\_OR\_NULL\_PTR(\*tmp1)) goto error;

(\*tmp1)->next = NULL;

(\*tmp1)->start\_time = ktime\_get();

pr\_info("Hello, World!\n");

(\*tmp1)->finish\_time = ktime\_get();

tmp2 = tmp1;

tmp1 = &(\*tmp1)->next;

}

return 0;

error:

pr\_err("Out of memory!\n");

free\_list(0);

return -ENOMEM;

}

EXPORT\_SYMBOL(hello\_print);

static int \_\_init hello\_init(void) {

return 0;

}

static void \_\_exit hello\_exit(void) {

free\_list(1);

}

module\_init(hello\_init);

module\_exit(hello\_exit);

**hello2.c**

#include <linux/init.h>

#include <linux/module.h>

#include <linux/moduleparam.h>

#include <linux/printk.h>

#include <hello1.h>

MODULE\_AUTHOR("Jack Shendrikov <jackshen@ukr.net>");

MODULE\_DESCRIPTION("Hello, world in Linux Kernel Training");

MODULE\_LICENSE("Dual BSD/GPL");

MODULE\_VERSION("2.0");

static uint count = 1;

module\_param(count, uint, S\_IRUGO);

MODULE\_PARM\_DESC(count, "Count to print 'Hello, world!'");

static int \_\_init hello\_init(void) {

return hello\_print(count);

}

static void \_\_exit hello\_exit(void) {

/\* do nothing \*/

}

module\_init(hello\_init);

module\_exit(hello\_exit);

**Makefile**

ccflags-y := -I$(PWD)/inc

ifneq ($(KERNELRELEASE),)

# kbuild part of makefile

obj-m := hello1.o hello2.o

else

# normal makefile

KDIR ?= /lib/modules/`uname -r`/build

default:

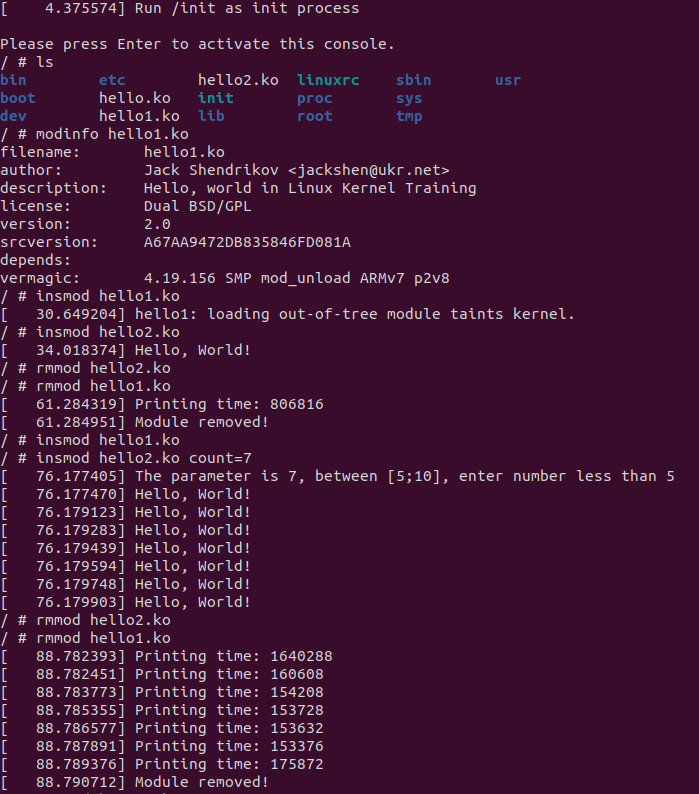
$(MAKE) -C $(KDIR) M=$$PWD

clean:

$(MAKE) -C $(KDIR) M=$$PWD clean

endif

**Приклади виконання**



Посилання на репозиторій: https://github.com/JackShen1/AK2\_Lab6