НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

ФАКУЛЬТЕТ ІНФОРМАТИКИ І ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

КАФЕДРА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

**Лабораторна робота №6**

з дисципліни **«**Архітектура комп’ютерів 2**»**

Виконав:

студент 3 курсу

групи ІВ-82

Сорока Андрій

Перевірив:

Нікольський С. С.

Київ 2020 р.

**Лістинг програми:**

**hello.c**

/\*

\* Copyright (c) 2017, GlobalLogic Ukraine LLC

\* All rights reserved.

\*

\* Redistribution and use in source and binary forms, with or without

\* modification, are permitted provided that the following conditions are met:

\* 1. Redistributions of source code must retain the above copyright

\* notice, this list of conditions and the following disclaimer.

\* 2. Redistributions in binary form must reproduce the above copyright

\* notice, this list of conditions and the following disclaimer in the

\* documentation and/or other materials provided with the distribution.

\* 3. All advertising materials mentioning features or use of this software

\* must display the following acknowledgement:

\* This product includes software developed by the GlobalLogic.

\* 4. Neither the name of the GlobalLogic nor the

\* names of its contributors may be used to endorse or promote products

\* derived from this software without specific prior written permission.

\*

\* THIS SOFTWARE IS PROVIDED BY GLOBALLOGIC UKRAINE LLC `AS IS` AND ANY

\* EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED

\* WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE

\* DISCLAIMED. IN NO EVENT SHALL GLOBALLOGIC UKRAINE LLC BE LIABLE FOR ANY

\* DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

\* (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;

\* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND

\* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

\* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS

\* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

\*/

#include <linux/init.h>

#include <linux/module.h>

#include <linux/printk.h>

#include <linux/slab.h>

#include <linux/ktime.h>

struct my\_list\_head {

struct my\_list\_head \*next;

ktime\_t time;

};

MODULE\_AUTHOR("Soroka Andrey");

MODULE\_DESCRIPTION("Lab6\_hello\_world\_output");

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

static struct my\_list\_head \*hd;

static uint count;

module\_param(count , uint , 0444);

MODULE\_PARM\_DESC(count , "variable for quantity of hello world output");

static int \_\_init hello(void)

{

uint i;

struct my\_list\_head \*first\_var, \*second\_var;

pr\_info("count: %d\n", count);

if (count == 0) {

pr\_warn("Parameter=0");

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

pr\_warn("Parameter is between 5 and 10");

} else if (count > 10) {

pr\_err("Parameter>10");

return -EINVAL;

}

hd = kmalloc(sizeof(struct my\_list\_head \*), GFP\_KERNEL);

first\_var = hd;

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

first\_var->next = kmalloc(sizeof(struct my\_list\_head), GFP\_KERNEL);

first\_var->time = ktime\_get();

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

second\_var = first\_var;

first\_var = first\_var->next;

}

if (count != 0) {

kfree(second\_var->next);

second\_var->next = NULL;

}

return 0;

}

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

{

struct my\_list\_head \*var;

while (hd != NULL && count != 0) {

var = hd;

pr\_info("Time: %lld", var->time);

hd = var->next;

kfree(var);

}

if (count == 0)

pr\_info("Time of output = 0 ");

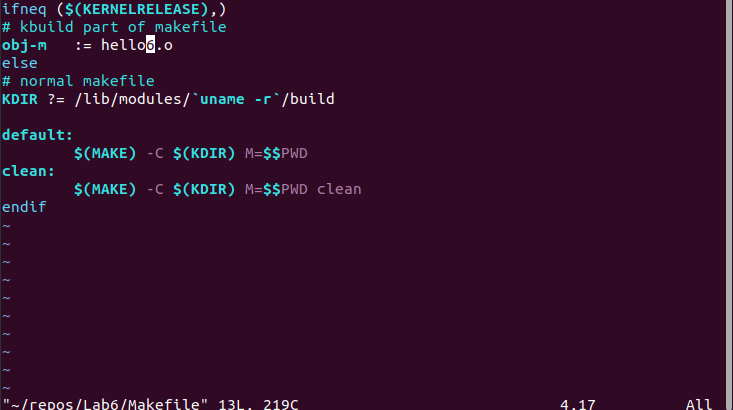
pr\_info("");

}

module\_init(hello);

module\_exit(hello\_exit);

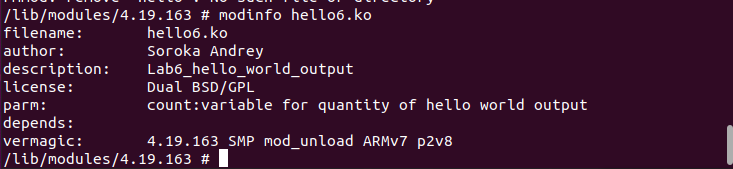
**MakeFile**



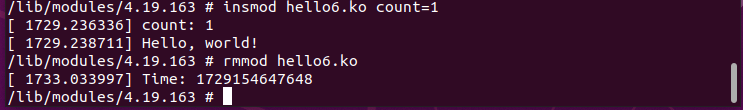
**Посилання на репозиторій:**

<https://github.com/AndreySoroka/Lab6>

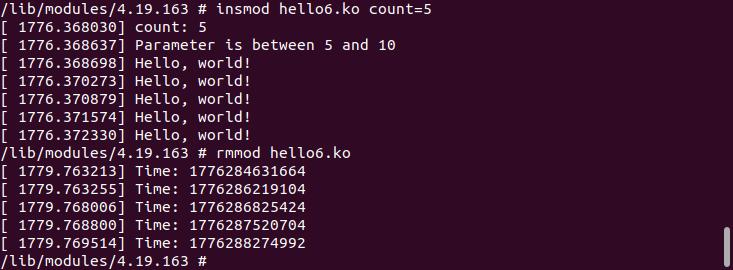
Приклад №1



Приклад №2



Приклад №3



Приклад №4

