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 ІО-81

Вигнанчук А.В.

Перевірив:

Каплунов А. В.

Київ 2020 р

**Результат роботи**







**Лістинг програми**

/\*

\* 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/kernel.h>

#include <linux/moduleparam.h>

#include <linux/printk.h>

#include <linux/types.h>

#include <linux/ktime.h>

#include <linux/slab.h>

struct head\_list {

struct head\_list \*next;

ktime\_t time;

};

MODULE\_AUTHOR("Serhii Popovych <serhii.popovych@globallogic.com>");

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

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

static struct head\_list \*head;

static int times = 1;

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

MODULE\_PARM\_DESC(times, "The amount of number to print hello world");

static int \_\_init initter(void)

{

uint i = 0;

struct head\_list \*key1, \*key2;

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

key1 = head;

if(times == 0) {

printk(KERN\_WARNING "The parameter is 0");

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

printk(KERN\_WARNING "The parameter is between 5 and 10");

} else if(times > 10) {

printk(KERN\_ERR "The parameter is greater than 10");

return -EINVAL;

}

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

key1->next = kmalloc(sizeof(struct head\_list), GFP\_KERNEL);

key1->time = ktime\_get();

pr\_info("Hello World!");

key2 = key1;

key1 = key1->next;

}

if (times != 0) {

kfree(key2->next);

key2->next = NULL;

}

return 0;

}

static void \_\_exit exitter(void)

{

struct head\_list \*key;

while (head != NULL && times != 0) {

key = head;

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

head = key->next;

kfree(key);

}

pr\_info("");

}

module\_init(initter);

module\_exit(exitter);