

МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ
КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Поволжский государственный университет телекоммуникаций и
информатики»

Факультет информатики и вычислительной техники

Кафедра При

ОТЧЁТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №2

“Работа с параметрами модуля ядра ОС GNU/Linux”

Дисциплина: Разработка драйверов

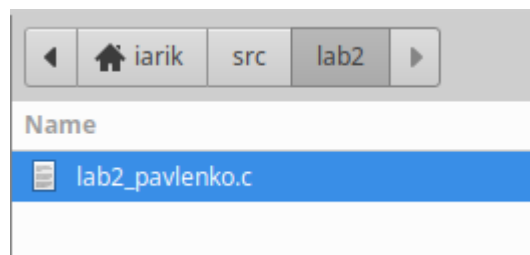
Выполнил: студент
При-21 Павленко Я.И.

Самара 2025

Цель работы: Знакомство с приемами программирования ОС GNU/Linux, создание простого модуля ядра системы.

Вариант 14ы

Задание 1. В рабочем каталоге пользователя создать каталог lab2. В данном каталоге создать файл исходного кода модуля ядра с именем lab2_ivanov.c (вместо ivanov вставить свою фамилию транслитерацией).



Задание 2. На основе результата выполнения предыдущей лабораторной работы, составить программу модуля ядра системы с двумя параметрами. Один параметр — целое число `int`, другой параметр — символьная строка `char`. При загрузке, модуль должен выводить строку, заданную во втором параметре указанное в первом параметре количество раз.

```
C lab2_pavlenko.c x M Makefile
C lab2_pavlenko.c
1  #include <linux/init.h>
2  #include <linux/module.h>
3  #include <linux/kernel.h>
4  #include <linux/types.h>
5  #include <linux/param.h>
6
7  MODULE_LICENSE("GPL");
8  MODULE_AUTHOR("Pavlenko Yaroslav Igorevich");
9  MODULE_DESCRIPTION("Lab2 (часть 2): модуль с параметрами howmany и whom");
10 MODULE_VERSION("0.01");
11
12
13
14 static int howmany = 1;
15 static char *whom = "world";
16
17
18 module_param(howmany, int, S_IRUGO);
19 MODULE_PARM_DESC(howmany, "Количество повторов сообщения");
20 |
21 module_param(whom, charp, S_IRUGO);
22 MODULE_PARM_DESC(whom, "Строка, которую нужно печатать");
23
24
```

```

24
25
26 static int __init lab2_init_pavlenko(void)
27 {
28     int i;
29
30     printk(KERN_INFO "=== lab2_pavlenko: init start ===\n");
31     printk(KERN_INFO "Параметр howmany = %d\n", howmany);
32     printk(KERN_INFO "Параметр whom    = %s\n", whom);
33
34
35     for (i = 0; i < howmany; ++i) {
36         printk(KERN_INFO "[%d] %s\n", i, whom);
37     }
38
39     printk(KERN_INFO "=== lab2_pavlenko: init end ===\n");
40     return 0;
41 }
42
43
44 static void __exit lab2_exit_pavlenko(void)
45 {
46     printk(KERN_INFO "=== lab2_pavlenko: exit ===\n");
47 }
48
49 module_init(lab2_init_pavlenko);
50 module_exit(lab2_exit_pavlenko);

```

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
● iarik@xUbuntu:~/src/lab2$ sudo make test
sudo dmesg -C
sudo insmod lab2_pavlenko.ko howmany=5 whom="PGUTI"
sudo rmmod lab2_pavlenko.ko
dmesg
[ 2391.982038] === lab2_pavlenko: init start ===
[ 2391.982042] Параметр howmany = 5
[ 2391.982043] Параметр whom    = PGUTI
[ 2391.982043] [0] PGUTI
[ 2391.982044] [1] PGUTI
[ 2391.982044] [2] PGUTI
[ 2391.982045] [3] PGUTI
[ 2391.982045] [4] PGUTI
[ 2391.982046] === lab2_pavlenko: init end ===
[ 2391.999567] === lab2_pavlenko: exit ===
○ iarik@xUbuntu:~/src/lab2$

```

Задание 3. В соответствии с таблицей вариантов, модифицировать программу.

4	Добавить параметр модуля, тип — булева величина. При значении параметра false при загрузке выводится только значения всех параметров модуля. Добавить массив параметров модуля (тип целочисленный, количество элементов - 7). Если первый элемент массива равен 1, вывести при загрузке модуля значения элементов массива в одну строку, через запятую.
---	---

```
C lab2_pavlenko.c
1  #include <linux/init.h>
2  #include <linux/module.h>
3  #include <linux/kernel.h>
4  #include <linux/types.h>
5  #include <linux/param.h>
6
7  MODULE_LICENSE("GPL");
8  MODULE_AUTHOR("Pavlenko Yaroslav Igorevich");
9  MODULE_DESCRIPTION("Lab2 (задание 2+3, вариант 4): howmany, whom, show_all(bool), arr[7]");
10 MODULE_VERSION("0.01");
11
12
13 static int howmany = 1;
14 module_param(howmany, int, S_IRUGO);
15 MODULE_PARM_DESC(howmany, "Сколько раз печатать строку whom");
16
17
18 static char *whom = "world";
19 module_param(whom, charp, S_IRUGO);
20 MODULE_PARM_DESC(whom, "Строка, которую нужно печатать");
21
22 static bool show_all = true;
23 module_param(show_all, bool, S_IRUGO);
24 MODULE_PARM_DESC(show_all, "Если false, печатать только все параметры. Если true, печатать howmany/whom + массив при услов");
25
26
27 static int arr[7] = {0, 0, 0, 0, 0, 0, 0};
28 static int arr_count = 0;
29 module_param_array(arr, int, &arr_count, S_IRUGO);
30 MODULE_PARM_DESC(arr, "Массив из 7 целых чисел. Если arr[0]==1, печатаем все arr[] через запятую");
31
32
33
```

```

67     if (arr[0] == 1) {
68         pos = 0;
69         pos += snprintf(buf + pos, sizeof(buf) - pos, "arr values: ");
70
71         for (i = 0; i < 7; ++i) {
72             if (i < 6)
73                 pos += snprintf(buf + pos, sizeof(buf) - pos, "%d, ", arr[i]);
74             else
75                 pos += snprintf(buf + pos, sizeof(buf) - pos, "%d", arr[i]);
76
77             if (pos >= sizeof(buf) - 20)
78                 break;
79         }
80         printk(KERN_INFO "%s\n", buf);
81     }
82
83     printk(KERN_INFO "=== lab2_pavlenko INIT: end ===\n");
84     return 0;
85 }
86
87 |
88 static void __exit lab2_exit_pavlenko(void)
89 {
90     printk(KERN_INFO "=== lab2_pavlenko: EXIT ===\n");
91 }
92
93 module_init(lab2_init_pavlenko);
94 module_exit(lab2_exit_pavlenko);

```

```

34 static int __init lab2_init_pavlenko(void)
35 {
36     int i;
37     char buf[256];
38     int pos = 0;
39
40     printk(KERN_INFO "=== lab2_pavlenko INIT: start ===\n");
41
42     printk(KERN_INFO "Параметр howmany = %d\n", howmany);
43     printk(KERN_INFO "Параметр whom = %s\n", whom);
44     printk(KERN_INFO "Параметр show_all = %s\n", show_all ? "true" : "false");
45     printk(KERN_INFO "Параметр arr_count = %d\n", arr_count);
46     for (i = 0; i < arr_count && i < 7; ++i) {
47         printk(KERN_INFO "  arr[%d] = %d\n", i, arr[i]);
48     }
49
50     for (; i < 7; ++i) {
51         printk(KERN_INFO "  arr[%d] = %d (по умолчанию)\n", i, arr[i]);
52     }
53
54
55     if (!show_all) {
56         printk(KERN_INFO "show_all == false → завершаем init\n");
57         printk(KERN_INFO "=== lab2_pavlenko INIT: end ===\n");
58         return 0;
59     }
60
61     printk(KERN_INFO "show_all == true → печатаем whom/howmany\n");
62     for (i = 0; i < howmany; ++i) {
63         printk(KERN_INFO "[%d] %s\n", i, whom);
64     }
65
66

```

```
● iarik@xUbuntu:~/src/lab2$ sudo make test
sudo dmesg -C
sudo insmod lab2_pavlenko.ko howmany=5 whom="PGUTI"
sudo rmmod lab2_pavlenko.ko
dmesg
[ 5556.696190] === lab2_pavlenko INIT: start ===
[ 5556.696193] Параметр howmany = 5
[ 5556.696194] Параметр whom = PGUTI
[ 5556.696195] Параметр show_all = true
[ 5556.696196] Параметр arr_count = 0
[ 5556.696196] arr[0] = 0 (по умолчанию)
[ 5556.696197] arr[1] = 0 (по умолчанию)
[ 5556.696197] arr[2] = 0 (по умолчанию)
[ 5556.696198] arr[3] = 0 (по умолчанию)
[ 5556.696198] arr[4] = 0 (по умолчанию)
[ 5556.696199] arr[5] = 0 (по умолчанию)
[ 5556.696199] arr[6] = 0 (по умолчанию)
[ 5556.696200] show_all == true → печатаем whom/howmany
[ 5556.696200] [0] PGUTI
[ 5556.696201] [1] PGUTI
[ 5556.696201] [2] PGUTI
[ 5556.696201] [3] PGUTI
[ 5556.696202] [4] PGUTI
[ 5556.696202] === lab2_pavlenko INIT: end ===
[ 5556.712927] === lab2_pavlenko: EXIT ===
○ iarik@xUbuntu:~/src/lab2$
```