

# **Отчет по выполнению лабораторной работы № 10**

**Дисциплина: Архитектура компьютера**

Ефремова Полина Александровна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
4.1	Задание для самостоятельной работы . . . . .	11
<b>5</b>	<b>Выводы</b>	<b>15</b>
	<b>Список литературы</b>	<b>16</b>

## Список иллюстраций

4.1	Создание файлов для выполнения лабораторной работы . . . . .	8
4.2	Ввод программы из листинга 10.1 . . . . .	9
4.3	Запуск . . . . .	9
4.4	Изменение прав доступа . . . . .	10
4.5	Изменение прав доступа . . . . .	11
4.6	Работа в соответствии с вариантом . . . . .	11
4.7	Работа в соответствии с вариантом . . . . .	12

## **Список таблиц**

# 1 Цель работы

Целью данной работы является приобретение навыков написания программ для работы с файлами.

## 2 Задание

1. Создание файлов в программах
2. Изменение прав на файлы для разных групп пользователей
3. Выполнение самостоятельных заданий по материалам лабораторной работы.

### 3 Теоретическое введение

ОС GNU/Linux является многопользовательской операционной системой. И для обеспечения защиты данных одного пользователя от действий других пользователей существуют специальные механизмы разграничения доступа к файлам. Кроме ограничения доступа, данный механизм позволяет разрешить другим пользователям доступ данным для совместной работы.

Права доступа определяют набор действий (чтение, запись, выполнение), разрешённых для выполнения пользователям системы над файлами. Для каждого файла пользователь может входить в одну из трех групп: владелец, член группы владельца, все остальные. Для каждой из этих групп может быть установлен свой набор прав доступа. Владелцем файла является его создатель

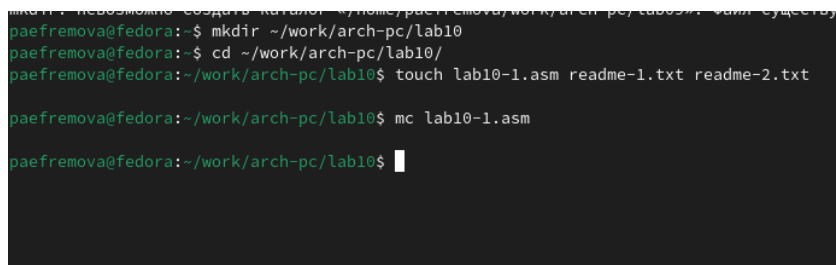
В операционной системе Linux существуют различные методы управления файлами, например, такие как создание и открытие файла, только для чтения или для чтения и записи, добавления в существующий файл, закрытия и удаления файла, предоставление прав доступа.

Обработка файлов в операционной системе Linux осуществляется за счет использования определенных системных вызовов. Для корректной работы и доступа к файлу при его открытии или создании, файлу присваивается уникальный номер (16-битное целое число) – дескриптор файла.

Общий алгоритм работы с системными вызовами в Nasm можно представить в следующем виде: 1. Поместить номер системного вызова в регистр EAX; 2. Поместить аргументы системного вызова в регистрах EBX, ECX и EDX; 3. Вызов прерывания (int 80h); 4. Результат обычно возвращается в регистр EAX.

## 4 Выполнение лабораторной работы

1. Создаю каталог для программ лабораторной работы № 10, перехожу в него и создаю файлы lab10-1.asm, readme-1.txt и readme-2.txt (рис. 4.1).



```
paefremova@fedora:~$ mkdir ~/work/arch-pc/lab10
paefremova@fedora:~$ cd ~/work/arch-pc/lab10/
paefremova@fedora:~/work/arch-pc/lab10$ touch lab10-1.asm readme-1.txt readme-2.txt
paefremova@fedora:~/work/arch-pc/lab10$ mc lab10-1.asm
paefremova@fedora:~/work/arch-pc/lab10$
```

Рис. 4.1: Создание файлов для выполнения лабораторной работы

2. В файл lab10-1.asm ввожу программу из листинга 10.1 (рис. 4.2).



```

GNU nano 7.2 /home/paefremova/work/arch-pc/lab10/lab10-1.asm
%include 'in_out.asm'
SECTION .data
filename db 'readme.txt', 0h ; имя файла
msg db 'Введите строку для записи в файл: ', 0h ; Сообщение
SECTION .bss
contents resb 255 ; переменная для вводимой строки
SECTION .text
global _start
_start:
; --- Печать сообщения 'msg'
mov eax,msg
call sprint
; --- Запись введенной с клавиатуры строки в 'contents'
mov ecx, contents
mov edx, 255
call sread
; --- Открытие существующего файла ('sys_open')
mov ecx, 2 ; открываем для записи (2)
mov ebx, filename
mov eax, 5
int 80h
; --- Запись дескриптора файла в 'esi'
mov esi, eax
; --- Расчет длины введенной строки
mov eax, contents ; в 'eax' запишется количество
call slen ; введенных байтов
; --- Записываем в файл 'contents' ('sys_write')
mov edx, eax
mov ecx, contents
mov ebx, esi
mov eax, 4
int 80h
; --- Закрываем файл ('sys_close')
mov ebx, esi
mov eax, 6
int 80h
call quit

```

Рис. 4.2: Ввод программы из листинга 10.1

### 3. Обработка и запуск программы. (рис. 4.3).

```

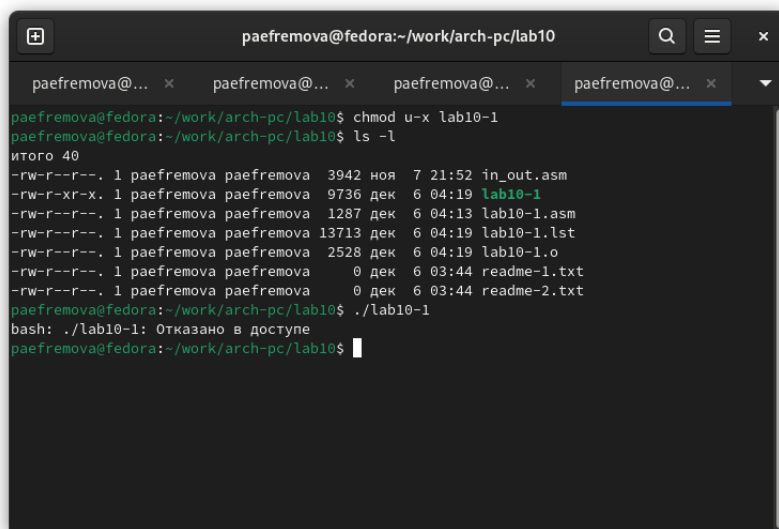
paefremova@fedora:~/work/arch-pc/lab10
paefremova@fedora:~/work/arch-pc/lab10$ nasm -f elf -g -l lab10-1.lst lab10-1.asm
paefremova@fedora:~/work/arch-pc/lab10$ ld -m elf_i386 -o lab10-1 lab10-1.o
paefremova@fedora:~/work/arch-pc/lab10$ ./lab10-1
Введите строку для записи в файл: polinariaaa
paefremova@fedora:~/work/arch-pc/lab10$ ls -l
итого 40
-rw-r--r--. 1 paefremova paefremova 3942 ноя 7 21:52 in_out.asm
-rwxr-xr-x. 1 paefremova paefremova 9736 дек 6 04:19 lab10-1
-rw-r--r--. 1 paefremova paefremova 1287 дек 6 04:13 lab10-1.asm
-rw-r--r--. 1 paefremova paefremova 13713 дек 6 04:19 lab10-1.lst
-rw-r--r--. 1 paefremova paefremova 2528 дек 6 04:19 lab10-1.o
-rw-r--r--. 1 paefremova paefremova 0 дек 6 03:44 readme-1.txt
-rw-r--r--. 1 paefremova paefremova 0 дек 6 03:44 readme-2.txt
paefremova@fedora:~/work/arch-pc/lab10$ cat readme-1.txt
polinariaaa
paefremova@fedora:~/work/arch-pc/lab10$

```

Рис. 4.3: Запуск

### 4. С помощью команды chmod изменю права доступа к исполняемому файлу

lab10-1, запретив его выполнение. Попытаюсь выполнить файл. Система отказывает в исполнении файла, т.к. я как владелец запретила самой себе же исполнять программу (рис. 4.4).



```
paefremova@fedora:~/work/arch-pc/lab10
paefremova@... x paefremova@... x paefremova@... x paefremova@... x
paefremova@fedora:~/work/arch-pc/lab10$ chmod u-x lab10-1
paefremova@fedora:~/work/arch-pc/lab10$ ls -l
итого 40
-rw-r--r--. 1 paefremova paefremova 3942 ноя 7 21:52 in_out.asm
-rw-r-xr-x. 1 paefremova paefremova 9736 дек 6 04:19 lab10-1
-rw-r--r--. 1 paefremova paefremova 1287 дек 6 04:13 lab10-1.asm
-rw-r--r--. 1 paefremova paefremova 13713 дек 6 04:19 lab10-1.lst
-rw-r--r--. 1 paefremova paefremova 2528 дек 6 04:19 lab10-1.o
-rw-r--r--. 1 paefremova paefremova 0 дек 6 03:44 readme-1.txt
-rw-r--r--. 1 paefremova paefremova 0 дек 6 03:44 readme-2.txt
paefremova@fedora:~/work/arch-pc/lab10$ ./lab10-1
bash: ./lab10-1: Отказано в доступе
paefremova@fedora:~/work/arch-pc/lab10$
```

Рис. 4.4: Изменение прав доступа

5. Добавляю к исходному файлу программы права владельцу на исполнение, исполняемый текстовый файл интерпретирует каждую строку как команду, так как ни одна из строк не является командой bash, программа абсолютно ничего не делает (рис. 4.5).

```

paefremova@fedora:~/work/arch-pc/lab10$ chmod u+x lab10-1.asm
paefremova@fedora:~/work/arch-pc/lab10$ ./lab10-1.asm
./lab10-1.asm: line 1: fg: no job control
./lab10-1.asm: line 3: SECTION: command not found
./lab10-1.asm: line 4: filename: command not found
./lab10-1.asm: line 5: msg: command not found
./lab10-1.asm: line 7: SECTION: command not found
./lab10-1.asm: line 8: contents: command not found
./lab10-1.asm: line 10: SECTION: command not found
./lab10-1.asm: line 11: global: command not found
./lab10-1.asm: line 12: _start:: command not found
./lab10-1.asm: line 13: mov: command not found
./lab10-1.asm: line 14: call: command not found
./lab10-1.asm: line 15: mov: command not found
./lab10-1.asm: line 16: mov: command not found
./lab10-1.asm: line 17: call: command not found
./lab10-1.asm: line 19: mov: command not found
./lab10-1.asm: line 20: mov: command not found
./lab10-1.asm: line 21: mov: command not found
./lab10-1.asm: line 22: int: command not found
./lab10-1.asm: line 24: mov: command not found
./lab10-1.asm: line 26: mov: command not found
./lab10-1.asm: line 27: call: command not found
./lab10-1.asm: line 29: mov: command not found
./lab10-1.asm: line 30: mov: command not found
./lab10-1.asm: line 31: mov: command not found
./lab10-1.asm: line 32: mov: command not found
./lab10-1.asm: line 33: int: command not found
./lab10-1.asm: line 35: mov: command not found
./lab10-1.asm: line 36: mov: command not found
./lab10-1.asm: line 37: int: command not found
./lab10-1.asm: line 39: call: command not found

```

Рис. 4.5: Изменение прав доступа

6. Согласно своему варианту, мне нужно установить соответствующие ему права на текстовые файлы, созданные в начале лабораторной работы:

В символьном виде для 1-го readme файла -x -w- r-x В двоичной системе для 2-го readme файла 001 101 010 Перевожу группу битов в восьмеричную систему, символьную запись подгоняю под синтаксис и получаю нужные аргументы для chmod (рис. 4.6).

```

# Введите строку для записи в файл: Ukjhgffyhj
paefremova@fedora:~/work/arch-pc/lab10$ chmod u=x,q=w,o=rx readme-1.txt
chmod: неверный режим: «u=x,q=w,o=rx»
По команде «chmod --help» можно получить дополнительную информацию.
paefremova@fedora:~/work/arch-pc/lab10$ chmod u=rx readme-1.txt
paefremova@fedora:~/work/arch-pc/lab10$ chmod g=w,o=rx readme-1.txt
paefremova@fedora:~/work/arch-pc/lab10$ chmod 152 readme-2.txt
paefremova@fedora:~/work/arch-pc/lab10$

```

Рис. 4.6: Работа в соответствии с вариантом

## 4.1 Задание для самостоятельной работы

Пишу программу, транслирую и компилирую. Программа должна выводить приглашение, просить ввод с клавиатуры и создавать текстовый файл с указан-

ной в программе строкой и вводом пользователя. Запускаю программу, проверяю наличие и содержание созданного текстового файла, программа работает корректно. рис. 4.7).

```
paefremova@fedora:~/work/arch-pc/lab10$ touch lab10-2.asm
paefremova@fedora:~/work/arch-pc/lab10$ mousepad lab10-2.asm
paefremova@fedora:~/work/arch-pc/lab10$ nasm -f elf lab10-2.asm
paefremova@fedora:~/work/arch-pc/lab10$ ld -m elf_i386 -o lab10-2 lab10-2.o
paefremova@fedora:~/work/arch-pc/lab10$ ./lab10-2
Как Вас зовут?Polina
paefremova@fedora:~/work/arch-pc/lab10$ ls
in_out.asm  lab10-1.asm  lab10-1.o  lab10-2.asm  name.txt      readme-2.txt
lab10-1     lab10-1.lst  lab10-2   lab10-2.o    readme-1.txt
paefremova@fedora:~/work/arch-pc/lab10$ cat name.txt
Меня зовут Polina
paefremova@fedora:~/work/arch-pc/lab10$
```

Рис. 4.7: Работа в соответствии с вариантом

Ниже прикрепляю листинг программы:

```
%include 'in_out.asm'
```

```
SECTION .data
```

```
filename db 'name.txt', 0
```

```
prompt db 'Как Вас зовут?', 0
```

```
intro db 'Меня зовут ', 0
```

```
SECTION .bss
```

```
name resb 255
```

```
SECTION .text
```

```
global _start
```

```
_start:
```

```
mov eax, prompt
```

```
call sprint
```

```
mov ecx, name
```

```
mov edx, 255
```

```
call sread
```

```
mov eax, 8
```

```
mov ebx, filename
```

```
mov ecx, 0744o
```

```
int 80h
```

```
mov esi, eax
```

```
mov eax, intro
```

```
call slen
```

```
mov edx, eax
```

```
mov ecx, intro
```

```
mov ebx, esi
```

```
mov eax, 4
```

```
int 80h
```

```
mov eax, name
```

```
call slen
```

```
mov edx, eax
```

```
mov ecx, name
```

```
mov ebx, esi
```

```
mov eax, 4
```

```
int 80h
```

```
mov ebx, esi
```

```
mov eax, 6
```

```
int 80h
```

**call** quit

## **5 Выводы**

В процессе выполнения лабораторной работы я приобрела навыки написания программ для работы с файлами, научилась редактировать права для файлов.

# Список литературы

1. Курс на ТУИС
2. Программирование на языке ассемблера NASM Столяров А. В. ::: {#refs} :::