

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра фундаментальной информатики и информационных технологий

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №10

Тема: Работа с файлами средствами Nasm

Дисциплина: Архитектура компьютеров

Студент: Герчет Вячеслав

Группа: НКАбд-03-25

Студ. билет № 1132255650

Преподаватель: Штепа Кристина Александровна

МОСКВА

2025 г.

## **Содержание**

1. Цель работы
2. Задание
3. Теоретическое введение
4. Выполнение лабораторной работы
5. Выполнение самостоятельной работы
6. Выводы
7. Список литературы

## Список иллюстраций

3.1 Создаем файлы с помощью команды touch

3.2 Заполняем файл

3.3 Запускаем файл и проверяем его работу

3.4 Используем команду chmod для установки нужных прав, после это-го пытаемся запустить файл

3.5 Используем команду chmod для установки нужных прав, после это-го пытаемся запустить файл

3.6 Используем команду chmod для установки нужных прав, после это-го проверяем правильность выполнения командой ls -l

3.7 Создаем файл командой touch

3.8 Пишем программу в midnight commander

3.9 Проверяем работу программы

## 1. Цель работы

Целью данной лабораторной работы является приобретение навыков написания программ для работы с файлами на языке ассемблера NASM, а также изучение управления правами доступа к файлам в операционной системе GNU/Linux.

## 2. Задание

Задание: написать программы на NASM для работы с файлами в Linux (создание, открытие, чтение, запись), изменить права доступа с помощью chmod и реализовать запись введённого имени пользователя в файл name.txt.

## 3. Теоретическое введение

Файловая система в операционных системах семейства GNU/Linux использует механизм прав доступа для защиты данных и разграничения прав пользователей. Каждый файл имеет владельца, группу и набор прав для трёх категорий пользователей: владельца, группы и остальных. Права определяют возможность чтения, записи и выполнения файла и могут устанавливаться как в символьном, так и в числовом (восьмеричном) виде с помощью команды chmod.

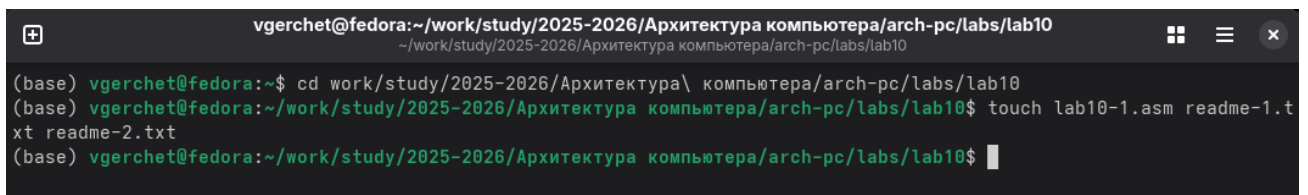
Для работы с файлами на низком уровне в языке ассемблера NASM используются системные вызовы операционной системы. К основным системным вызовам относятся sys\_creat,

`sys_open`, `sys_read`, `sys_write`, `sys_close`, `sys_lseek` и `sys_unlink`, которые позволяют создавать, открывать, читать, записывать, изменять и удалять файлы. Все операции с файлами выполняются через файловые дескрипторы -- уникальные числовые идентификаторы, возвращаемые при открытии или создании файла.

Системные вызовы выполняются через прерывание `int 80h`, при этом номер вызова передаётся через регистр `EAX`, а аргументы — через регистры `EBX`, `ECX` и `EDX`. Такой механизм обеспечивает взаимодействие программы с ядром операционной системы и позволяет реализовывать файловый ввод и вывод на уровне машинных инструкций.

## 4. Выполнение лабораторной работы

Создаем файл (рис. 3.1).

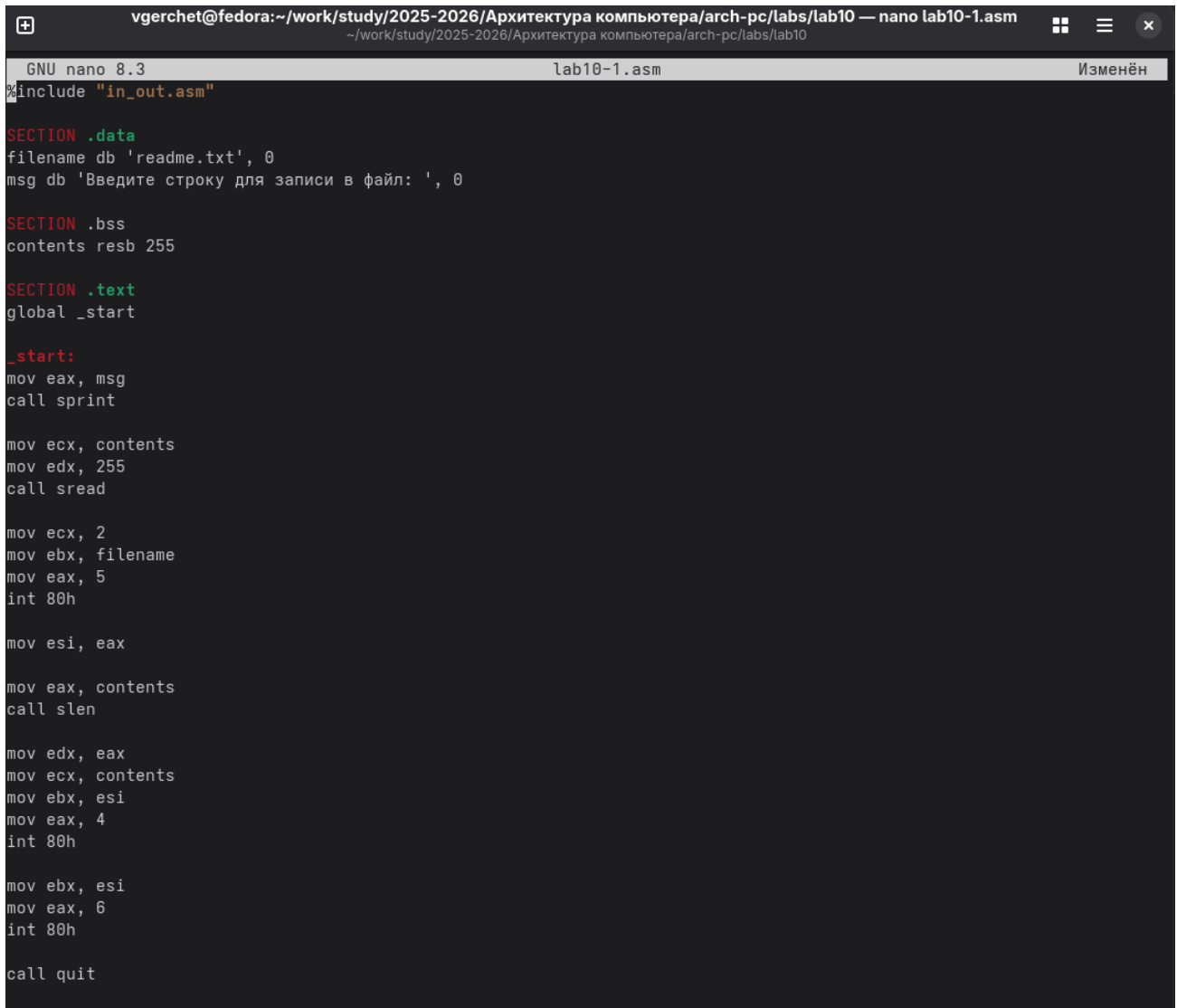


```
vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10
~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10
(base) vgerchet@fedora:~$ cd work/study/2025-2026/Архитектура\ компьютера/arch-pc/labs/lab10
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$ touch lab10-1.asm readme-1.txt
readme-2.txt
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$
```

Рис. 3.1: Создаем файл с помощью команды `touch`

Открываем файл в Midnight Commander и заполняем его в соответствии с

листингом 10.1 (рис. 3.2).



```
GNU nano 8.3 lab10-1.asm Изменён
~/.work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10

%include "in_out.asm"

SECTION .data
filename db 'readme.txt', 0
msg db 'Введите строку для записи в файл: ', 0

SECTION .bss
contents resb 255

SECTION .text
global _start

_start:
mov eax, msg
call sprint

mov ecx, contents
mov edx, 255
call sread

mov ecx, 2
mov ebx, filename
mov eax, 5
int 80h

mov esi, eax

mov eax, contents
call slen

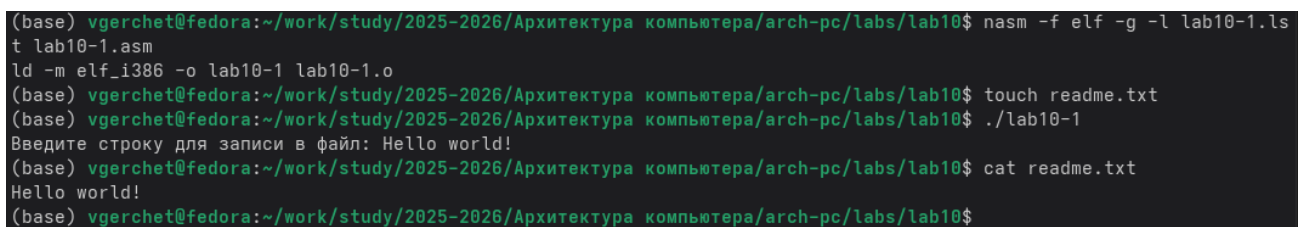
mov edx, eax
mov ecx, contents
mov ebx, esi
mov eax, 4
int 80h

mov ebx, esi
mov eax, 6
int 80h

call quit
```

Рис. 3.2: Заполняем файл

Создаем исполняемый файл и запускаем его (рис. 3.3).



```
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$ nasm -f elf -g -l lab10-1.ls
t lab10-1.asm
ld -m elf_i386 -o lab10-1 lab10-1.o
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$ touch readme.txt
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$ ./lab10-1
Введите строку для записи в файл: Hello world!
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$ cat readme.txt
Hello world!
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$
```

Рис. 3.3: Запускаем файл и смотрим на его работу

Изменяем права доступа к файлу, запретив его выполнение. Пробуем запустить файл(рис. 3.4)

```
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$ ./lab10-1
bash: ./lab10-1: Отказано в доступе
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$
```

Рис. 3.4: Используем команду `chmod` для установки нужных прав, после этого пытаемся запустить файл

Выдало: отказано в доступе. Значит мы поставили правильный запрет на выполнение.

Изменяем права доступа к файлу с исходным текстом программы, добавив права на исполнение. Пробуем запустить файл (рис. 3.5).

```
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$ chmod u+x lab10-1.asm
ls -l lab10-1.asm
-rwxr--r--. 1 vgerchet vgerchet 477 дек  9 13:38 lab10-1.asm
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$ ./lab10-1.asm
./lab10-1.asm: строка 1: fg: нет управления заданиями
./lab10-1.asm: строка 3: SECTION: команда не найдена
./lab10-1.asm: строка 4: filename: команда не найдена
./lab10-1.asm: строка 5: msg: команда не найдена
./lab10-1.asm: строка 7: SECTION: команда не найдена
./lab10-1.asm: строка 8: contents: команда не найдена
./lab10-1.asm: строка 10: SECTION: команда не найдена
./lab10-1.asm: строка 11: global: команда не найдена
./lab10-1.asm: строка 13: _start:: команда не найдена
./lab10-1.asm: строка 14: mov: команда не найдена
./lab10-1.asm: строка 15: call: команда не найдена
./lab10-1.asm: строка 17: mov: команда не найдена
./lab10-1.asm: строка 18: mov: команда не найдена
./lab10-1.asm: строка 19: call: команда не найдена
./lab10-1.asm: строка 21: mov: команда не найдена
./lab10-1.asm: строка 22: mov: команда не найдена
./lab10-1.asm: строка 23: mov: команда не найдена
./lab10-1.asm: строка 24: int: команда не найдена
./lab10-1.asm: строка 26: mov: команда не найдена
./lab10-1.asm: строка 28: mov: команда не найдена
./lab10-1.asm: строка 29: call: команда не найдена
./lab10-1.asm: строка 31: mov: команда не найдена
./lab10-1.asm: строка 32: mov: команда не найдена
./lab10-1.asm: строка 33: mov: команда не найдена
./lab10-1.asm: строка 34: mov: команда не найдена
./lab10-1.asm: строка 35: int: команда не найдена
./lab10-1.asm: строка 37: mov: команда не найдена
./lab10-1.asm: строка 38: mov: команда не найдена
./lab10-1.asm: строка 39: int: команда не найдена
./lab10-1.asm: строка 41: call: команда не найдена
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$
```

Рис. 3.5: Используем команду `chmod` для установки нужных прав, после этого пытаемся запустить файл

lab10-1.asm является файлом с исходным кодом программы на языке ассемблера, искусственно добавление права на исполнение не даст ожидаемого результата. Такие файлы нужно компилировать или ассемблировать в машинный код, а затем выполнять.

## 5. Выполнение самостоятельной работы

### ВАРИАНТ 11

Предоставляем права доступа к 2ум файлам, согласно варианту 11 в символьном и двоичном виде, затем проверяем работу команд. (рис. 3.6).

```
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$ chmod u=x,g=r,o=w readme-1.txt
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$ chmod 047 readme-2.txt
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$ ls -l readme-1.txt readme-2.txt
---xr---w-. 1 vgerchet vgerchet 0 дек  9 13:36 readme-1.txt
----r--rwx. 1 vgerchet vgerchet 0 дек  9 13:36 readme-2.txt
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$
```

Рис. 3.6: Используем команду chmod для установки нужных прав, после этого проверяем правильность выполнения командой ls -l

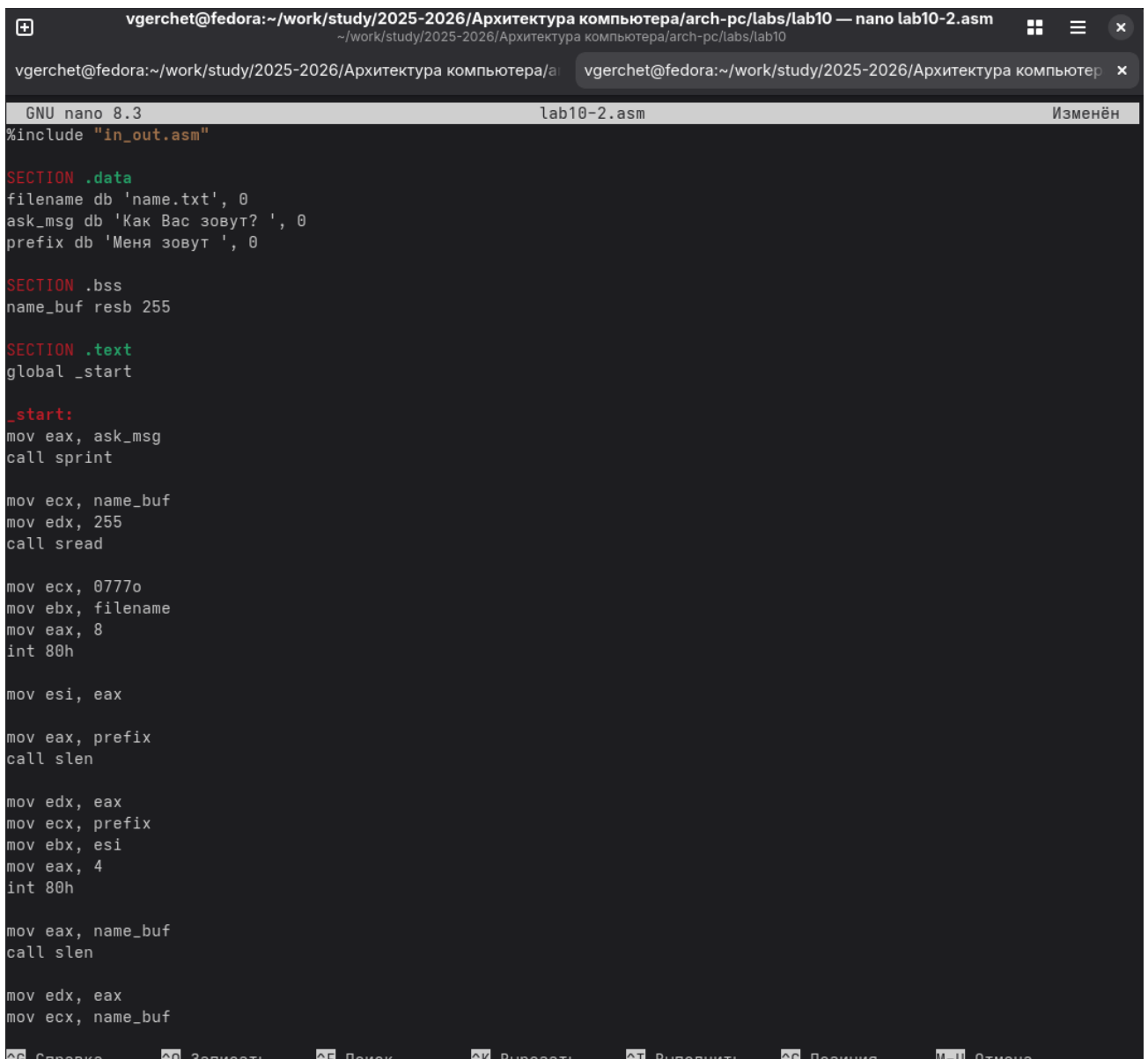
## 5. Выполнение самостоятельной работы

Создаем новый файл (рис. 3.7).

```
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$ touch lab10-2.asm
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$
```

Рис. 3.7: Создаем файл командой touch

Пишем программу, которая выполнит представленный список действий(рис. 3.8).



```
GNU nano 8.3 lab10-2.asm Изменён
#include "in_out.asm"

SECTION .data
filename db 'name.txt', 0
ask_msg db 'Как Вас зовут? ', 0
prefix db 'Меня зовут ', 0

SECTION .bss
name_buf resb 255

SECTION .text
global _start

_start:
mov eax, ask_msg
call sprint

mov ecx, name_buf
mov edx, 255
call sread

mov ecx, 0777o
mov ebx, filename
mov eax, 8
int 80h

mov esi, eax

mov eax, prefix
call slen

mov edx, eax
mov ecx, prefix
mov ebx, esi
mov eax, 4
int 80h

mov eax, name_buf
call slen

mov edx, eax
mov ecx, name_buf
```

Рис. 3.8: Пишем программу в midnight commander

Создаем исполняемый файл и запускаем его, после этого проверяем создался



ли новый файл, затем смотрим, как он заполнен (рис. 3.9).

```
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$ nasm -f elf -g -l lab10-2.lst lab10-2.asm
ld -m elf_i386 -o lab10-2 lab10-2.o
./lab10-2
Как Вас зовут? Герчет Вячеслав
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$ ls -l name.txt
cat name.txt
-rwxr-xr-x. 1 vgerchet vgerchet 50 дек  9 13:56 name.txt
Меня зовут Герчет Вячеслав
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab10$
```

Рис. 3.9: Проверяем работу программы

## 4. Выводы

В ходе выполнения лабораторной работы я научился использовать системные вызовы для чтения и записи данных, а также управлять правами доступа к файлам с помощью команды `chmod`. Полученные знания позволили лучше понять принципы работы файловой системы Linux и процесс взаимодействия программ с операционной системой.

## 5. Список литературы

NASM Documentation. <https://www.nasm.us/docs.html>

Демидова А. В. Архитектура ЭВМ. Лабораторная работа №9

GDB: The GNU Project Debugger. <https://sourceware.org/gdb>

OpenNet. Расширенный ассемблер NASM. — URL: <https://www.opennet.ru/docs/RUS/nasm/>