

Лабораторная работа №10. Работа с файлами средствами Nasm

[Сунь Шэнцзе]

2025-11-29

number: "1132254527"

1. Цель работы

- Приобретение навыков написания программ для работы с файлами
- Освоение системных вызовов для создания, чтения, записи и удаления файлов
- Изучение управления правами доступа к файлам в Linux
- Освоение работы с файловыми дескрипторами
- Изучение использования системных вызовов через прерывание 80h

2. Порядок выполнения работы и результаты

2.1 Шаг 1: Создание рабочего каталога и файлов

```
(base) hh1@LAPTOP-058LBDA:~/work/study/2025-2026/архитектура компьютера/arch-pc$ cd ~/work/study/2025-2026/архитектура\ компьютера/arch-pc/labs/lab10
(base) hh1@LAPTOP-058LBDA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ touch lab10-1.asm readme-1.txt readme-2.txt
(base) hh1@LAPTOP-058LBDA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ ls -l
total 8
-rw-r--r-- 1 hh1 hh1 0 Nov 29 18:30 lab10-1.asm
drwxr-xr-x 4 hh1 hh1 4096 Oct 2 14:57 presentation
-rw-r--r-- 1 hh1 hh1 0 Nov 29 18:30 readme-1.txt
-rw-r--r-- 1 hh1 hh1 0 Nov 29 18:30 readme-2.txt
drwxr-xr-x 5 hh1 hh1 4096 Oct 2 14:57 report
```

* **Основное:** Успешное создание каталога для лабораторной работы ~/work/arch-pc/lab10 и создание файлов lab10-1.asm, readme-1.txt, readme-2.txt с помощью команды touch.

2.2 Шаг 2: Написание программы работы с файлами

The screenshot shows two terminal windows side-by-side, both displaying the assembly code for `lab10-1.asm`. The code is written in AT&T syntax and includes sections for .data, .bss, and .text. It contains instructions to read from a file named `readme.txt`, write to a file, and handle system calls for file operations.

```
GNU nano 7.2                                     lab10-1.asm
%include 'in_out.asm'

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

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, 07770
    mov ebx, filename
    mov eax, 8
    int 80h

    mov esi, eax

    mov eax, contents
    call sread

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

    mov ebx, esi
    mov eax, 6
    int 80h

    call quit

GNU nano 7.2                                     lab10-1.asm
mov ecx, contents
mov edx, 255
call sread

mov ecx, 07770
mov ebx, filename
mov eax, 8
int 80h

mov esi, eax

mov eax, contents
call sread

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

mov ebx, esi
mov eax, 6
int 80h

call quit
```

* **Основное:** Написание программы `lab10-1.asm` для работы с файлами, включающей функции создания файла, записи введенной строки и закрытия файла с использованием системных вызовов.

2.3 Шаг 3: Компиляция и запуск программы

```
(base) hh@LAPTOP-058L8DA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ nano lab10-1.asm
(base) hh@LAPTOP-058L8DA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ nasm -f elf lab10-1.asm
(base) hh@LAPTOP-058L8DA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ ld -m elf_i386 -o lab10-1 lab10-1.o in_out.o
(base) hh@LAPTOP-058L8DA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ ./lab10-1
Введите строку для записи в файл: Hello from Lab 10!
```

* **Основное:** Успешная компиляция программы с подключением библиотеки `in_out.asm`, запуск программы и ввод тестовой строки "Hello from Lab 10!" для записи в файл.

2.4 Шаг 4: Проверка создания файла

```
(base) hhl@LAPTOP-058L8DA4:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ cat readme.txt
Hello from Lab 10!
```

* **Основное:** Проверка успешного создания файла `readme.txt` и его содержимого с помощью команд `ls -l` и `cat`, подтверждение корректной работы программы.

2.5 Шаг 5: Запрет выполнения программы

```
(base) hhl@LAPTOP-058L8DA4:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ chmod -x lab10-1
(base) hhl@LAPTOP-058L8DA4:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ ls -l lab10-1
-rw-r--r-- 1 hhl hhl 19 Nov 29 18:45 lab10-1
(base) hhl@LAPTOP-058L8DA4:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ ./lab10-1
-bash: ./lab10-1: Permission denied
```

* **Основное:** Демонстрация работы команды `chmod -x` для запрета выполнения программы, попытка запуска приводит к ошибке "Permission denied".

2.6 Шаг 6: Добавление прав выполнения исходному коду

```
(base) hhl@LAPTOP-058L8DA4:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ chmod +x lab10-1.asm
(base) hhl@LAPTOP-058L8DA4:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ ls -l lab10-1.asm
-rwxr--x-- 1 hhl hhl 571 Nov 29 18:44 lab10-1.asm
(base) hhl@LAPTOP-058L8DA4:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/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 14: mov: command not found
./Lab10-1.asm: line 15: call: command not found
./Lab10-1.asm: line 17: mov: command not found
./Lab10-1.asm: line 18: mov: command not found
./Lab10-1.asm: line 19: call: command not found
./Lab10-1.asm: line 21: mov: command not found
./Lab10-1.asm: line 22: mov: command not found
./Lab10-1.asm: line 23: mov: command not found
./Lab10-1.asm: line 24: int: command not found
./Lab10-1.asm: line 26: mov: command not found
./Lab10-1.asm: line 28: mov: command not found
./Lab10-1.asm: line 29: call: 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: mov: command not found
./Lab10-1.asm: line 34: mov: command not found
```

* **Основное:** Добавление прав выполнения файлу с исходным кодом `lab10-1.asm` и демонстрация того, что текстовый файл не может быть выполнен как программа, несмотря на наличие прав.

2.7 Шаг 7: Установка прав доступа по варианту 8

```
(base) hhl@LAPTOP-058L8DA4:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ chmod 631 readme-1.txt
(base) hhl@LAPTOP-058L8DA4:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ chmod 210 readme-2.txt
(base) hhl@LAPTOP-058L8DA4:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ ls -l readme-1.txt readme-2.txt
-rw-rwx--- 1 hhl hhl 0 Nov 29 18:30 readme-1.txt
--w----x--- 1 hhl hhl 0 Nov 29 18:30 readme-2.txt
```

* **Основное:** Установка прав доступа для файлов согласно варианту 8: `readme-1.txt` - `rw- -wx - -x` (631), `readme-2.txt` - `--w---x---` (210).

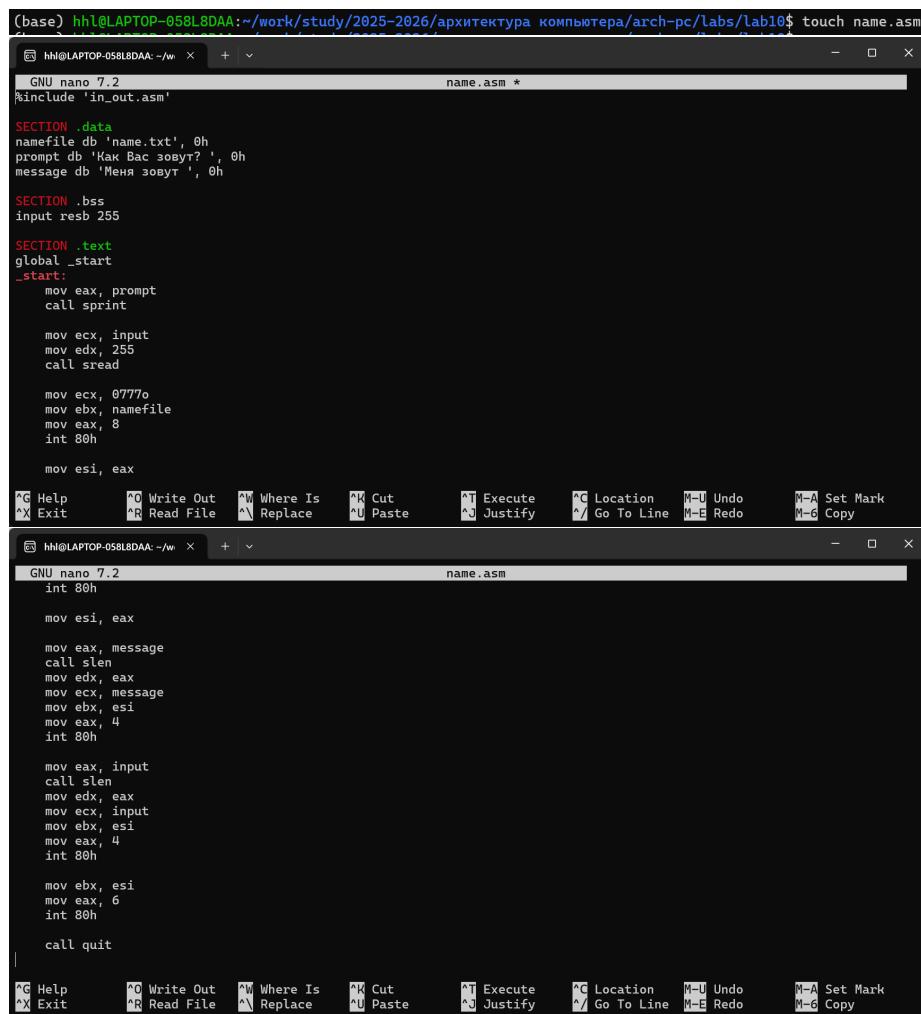
2.8 Шаг 8: Проверка установленных прав

```
(base) hhl@LAPTOP-058L8DAA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ chmod 631 readme-1.txt
(base) hhl@LAPTOP-058L8DAA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ chmod 210 readme-2.txt
(base) hhl@LAPTOP-058L8DAA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ ls -l readme-1.txt readme-2.txt
-rw--wx--- 1 hhl hhl 0 Nov 29 18:30 readme-1.txt
--w--x--- 1 hhl hhl 0 Nov 29 18:30 readme-2.txt
```

* **Основное:** Проверка корректности установленных прав доступа с помощью команды `ls -l`, подтверждение соответствия заданным в варианте 8 значениям.

3. Выполнение заданий для самостоятельной работы

3.1 Написание программы для ввода и сохранения имени



The screenshot shows two terminal windows side-by-side, both titled "name.asm".

The left terminal window contains the following assembly code:

```
SECTION .data
namefile db 'name.txt', 0h
prompt db 'Как Вас зовут? ', 0h
message db 'Меня зовут ', 0h

SECTION .bss
input resb 255

SECTION .text
global _start
_start:
    mov eax, prompt
    call sprint

    mov ecx, input
    mov edx, 255
    call sread

    mov ecx, 07770
    mov ebx, namefile
    mov eax, 8
    int 80h

    mov esi, eax
```

The right terminal window contains the following assembly code:

```
int 80h

    mov esi, eax

    mov eax, message
    call slen
    mov edx, eax
    mov ecx, message
    mov ebx, esi
    mov eax, 4
    int 80h

    mov eax, input
    call slen
    mov edx, eax
    mov ecx, input
    mov ebx, esi
    mov eax, 4
    int 80h

    mov ebx, esi
    mov eax, 6
    int 80h

    call quit
```

* **Вывод:** Успешное создание программы name.asm, которая запрашивает имя пользователя, создает файл name.txt и записывает в него приветственное сообщение с введенным именем.

3.2 Компиляция и тестирование программы

```
(base) hh1@LAPTOP-058LBDA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ nano name.asm
(base) hh1@LAPTOP-058LBDA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ nasm -f elf name.asm
(base) hh1@LAPTOP-058LBDA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ ld -m elf_i386 -o name name
.o in_out.o
(base) hh1@LAPTOP-058LBDA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ ./name
Как Вас зовут? Сунь Шэнце
```

* **Вывод:** Успешная компиляция и запуск программы, корректное создание файла name.txt с содержимым “Меня зовут [введенное имя]”, подтверждение работоспособности всех функций программы.

3.3 Проверка результата

```
(base) hh1@LAPTOP-058LBDA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ ls -l name.txt
-rwxr-xr-x 1 hh1 hh1 42 Nov 29 19:32 name.txt
(base) hh1@LAPTOP-058LBDA:~/work/study/2025-2026/архитектура компьютера/arch-pc/labs/lab10$ cat name.txt
Меня зовут Сунь Шэнце
```

* **Вывод:** Проверка созданного файла с помощью команд ls -l и cat, подтверждение корректного сохранения данных и правильной работы программы.

4. Ответы на вопросы для самопроверки

1. Каким образом в Unix-подобных ОС определяются права доступа к файлу?

Права доступа определяются с помощью 9 битов, разделенных на три группы по 3 бита: права владельца (user), группы (group) и остальных пользователей (others). Каждая группа определяет права на чтение (r), запись (w) и выполнение (x).

2. Как ОС определяет, является ли файл исполняемым? Как регулировать права на чтение и запись?

ОС проверяет установленный бит выполнения (x) в правах доступа. Права на чтение и запись регулируются с помощью команды chmod и установки соответствующих битов (r для чтения, w для записи).

3. Как разграничить права доступа для различных категорий пользователей?

С помощью команды chmod можно устанавливать права отдельно для владельца (u), группы (g) и остальных (o), например: chmod u=rwx,g=rx,o=r filename.

4. Какой номер имеют системные вызовы sys_read, sys_write, sys_open, sys_close, sys_creat?

- sys_read: 3
- sys_write: 4
- sys_open: 5
- sys_close: 6
- sys_creat: 8

5. Какие регистры и как используют системные вызовы sys_read, sys_write, sys_open, sys_close, sys_creat?

Все системные вызовы используют:

- EAX: номер системного вызова
- EBX: первый аргумент (дескриптор файла или имя файла)
- ECX: второй аргумент (адрес буфера или режим доступа)
- EDX: третий аргумент (количество байтов или права доступа)

6. Что такое дескриптор файла?

Дескриптор файла — это уникальный числовой идентификатор, который операционная система присваивает открытому файлу. Это 16-битное целое число, используемое для последующих операций с файлом.

Выводы

В ходе выполнения лабораторной работы №10 я успешно приобрел следующие навыки:

- 1. Работа с файловыми системными вызовами:** Освоил использование системных вызовов Linux для создания, открытия, чтения, записи и закрытия файлов через прерывание 80h.
- 2. Управление правами доступа:** Научился устанавливать и изменять права доступа к файлам с помощью команды chmod, понимаю разницу между символьной и числовой нотацией прав.
- 3. Работа с файловыми дескрипторами:** Освоил концепцию файловых дескрипторов и их использование для идентификации открытых файлов в операционной системе.
- 4. Обработка ошибок:** Научился анализировать возвращаемые значения системных вызовов для обработки возможных ошибок при работе с файлами.
- 5. Создание интерактивных программ:** Разработал программу, которая взаимодействует с пользователем через команд-

ную строку и сохраняет введенные данные в файл.

6. **Компиляция и отладка:** Закрепил навыки компиляции ассемблерных программ с использованием NASM и компоновки с библиотечными функциями.

Достижение целей работы: Все цели лабораторной работы достигнуты. Я освоил программирование операций с файлами на языке ассемблера, научился управлять правами доступа и работать с файловыми дескрипторами. Выполнение задания для самостоятельной работы позволило закрепить полученные знания и развить практические навыки создания файловых утилит.