

# **Лабораторная работа номер 10.**

**Работа с файлами средствами Nasm**

Сорокин Кирилл

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Задание</b>	<b>5</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>6</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
4.1	Самостоятельная работа . . . . .	10
<b>5</b>	<b>Выводы</b>	<b>13</b>
	<b>Список литературы</b>	<b>14</b>

## Список иллюстраций

4.1	Создание файлов и директорий . . . . .	7
4.2	Текст первой программы . . . . .	8
4.3	Создание файлов программы . . . . .	9
4.4	Работа первой программы . . . . .	9
4.5	Отказано в доступе . . . . .	9
4.6	Вариант 1 . . . . .	9
4.7	Изменение прав доступа для readme-1.txt . . . . .	10
4.8	Изменение прав доступа для readme-2.txt . . . . .	10
4.9	Текст программы самостоятельной работы . . . . .	11
4.10	Выполнение самостоятельной работы . . . . .	12

# 1 Цель работы

Научиться писать программы на языке ассемблера для работы с файлами.

## **2 Задание**

Изучить приведённый материал на практике и выполнить самостоятельную работу.

### 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. Владелец файла является его создатель.

## 4 Выполнение лабораторной работы

Создадим необходимые для работы директории и файлы (рис. 4.1).

```
kvsorokin@dk3n60 ~ $ mkdir ~/work/arch-pc/lab10
kvsorokin@dk3n60 ~ $ cd ~/work/arch-pc/lab10
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ touch lab10-1.asm readme-1.txt
readme-2.txt
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $
```

Рис. 4.1: Создание файлов и директорий

Откроем файл lab10-1.asm и введём в него текст листинга 10.1(рис. 4.2).

```

;-----
; Запись в файл строки введенной на запрос
;-----
%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: Текст первой программы

Скомпилируем программу(рис. 4.3).



```

kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ gedit lab10-1.asm
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ nasm -f elf lab10-1.asm
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ ld -m elf_i386 -o lab10-1 lab10-1.o

```

Рис. 4.3: Создание файлов программы

Проверим работу программы(рис. 4.4).

```

kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ touch readme.txt
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ ./lab10-1
Введите строку для записи в файл: Hello
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ cat readme.txt
Hello

```

Рис. 4.4: Работа первой программы

Изменим права доступа для программы и увидим, что теперь ей отказано в доступе. Это потому, что заданной командой мы отключаем право доступа.(рис. 4.5).

```

kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ chmod a-x lab10-1
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ ls -l
итого 18
-rw-r--r-- 1 kvsorokin studsci 3942 ноя  8 12:47 in_out.asm
-rw-r--r-- 1 kvsorokin studsci 9164 дек 13 12:38 lab10-1
-rw-r--r-- 1 kvsorokin studsci 1310 дек 13 12:37 lab10-1.asm
-rw-r--r-- 1 kvsorokin studsci 1472 дек 13 12:38 lab10-1.o
-rw-r--r-- 1 kvsorokin studsci   0 дек 13 12:28 readme-1.txt
-rw-r--r-- 1 kvsorokin studsci   0 дек 13 12:28 readme-2.txt
-rw-r--r-- 1 kvsorokin studsci   6 дек 13 12:43 readme.txt
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ ./lab10-1
bash: ./lab10-1: Отказано в доступе

```

Рис. 4.5: Отказано в доступе

В соответствии с нашим вариантом зададим необходимые права доступа для файлов.(рис. 4.6).

```

1          --x  -wx  rwx          000 110 010

```

Рис. 4.6: Вариант 1

Изменим права доступа для readme-1.txt на -x -wx rwx, и удостоверимся, что всё верно(рис. 4.7).

```
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ chmod 137 readme-1.txt
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ ls -l
итого 18
-rw-r--r-- 1 kvsorokin studsci 3942 ноя  8 12:47 in_out.asm
-rw-r--r-- 1 kvsorokin studsci 9164 дек 13 12:38 lab10-1
-rw-r--r-- 1 kvsorokin studsci 1310 дек 13 12:37 lab10-1.asm
-rw-r--r-- 1 kvsorokin studsci 1472 дек 13 12:38 lab10-1.o
---x-wxrw 1 kvsorokin studsci   0 дек 13 12:28 readme-1.txt
-rw-r--r-- 1 kvsorokin studsci   0 дек 13 12:28 readme-2.txt
-rw-r--r-- 1 kvsorokin studsci   6 дек 13 12:43 readme.txt
```

Рис. 4.7: Изменение прав доступа для readme-1.txt

Изменим права доступа для readme-2.txt на 000 110 010, и по таблице удостоверимся, что всё верно (рис. 4.8).

```
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ chmod 062 readme-2.txt
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ ls -l
итого 18
-rw-r--r-- 1 kvsorokin studsci 3942 ноя  8 12:47 in_out.asm
-rw-r--r-- 1 kvsorokin studsci 9164 дек 13 12:38 lab10-1
-rw-r--r-- 1 kvsorokin studsci 1310 дек 13 12:37 lab10-1.asm
-rw-r--r-- 1 kvsorokin studsci 1472 дек 13 12:38 lab10-1.o
---x-wxrw 1 kvsorokin studsci   0 дек 13 12:28 readme-1.txt
----rw--w- 1 kvsorokin studsci   0 дек 13 12:28 readme-2.txt
-rw-r--r-- 1 kvsorokin studsci   6 дек 13 12:43 readme.txt
```

Рис. 4.8: Изменение прав доступа для readme-2.txt

## 4.1 Самостоятельная работа

Основываясь на значения курса и урока, напишем программу для записи имени в созданный файл. (к сожалению снимок экрана не отображает весь линстинг)(рис. 4.9).

```

#include 'in_out.asm'

section .data
    nameRequest: db "Как вас зовут? - ", 0
    filename: db "name.txt", 0
    iam: db "Меня зовут "
    iamLength: equ $-iam

section .bss
    name: resb 255

section .text
    global _start

_start:
    mov eax, nameRequest
    call sprint

    mov ecx, name
    mov edx, 255
    call sread

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

    call _openfile

    mov edx, iamLength
    mov ecx, iam
    mov ebx, eax
    mov eax, 4
    int 80h

    call _closefile

    call _openfile

    mov edx, 2
    mov ecx, 0
    mov ebx, eax
    mov eax, 19
    int 80h
    mov esi, eax
    ....

```

Рис. 4.9: Текст программы самостоятельной работы

Проверим работу программы и удостоверимся, что создан необходимый файл и в него записано ‘Меня зовут Сорокин Кирилл’

```

kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ touch lab10-s.asm
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ gedit lab10-s.asm
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ nasm -f elf lab10-s.asm
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ ld -m elf_i386 -o lab10-s l
ab10-s.o
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ ./lab10-s
Как вас зовут? - Сорокин Кирилл
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ ls
in_out.asm  lab10-1.o  lab10-s.o  readme-2.txt
lab10-1     lab10-s    name.txt   readme.txt
lab10-1.asm lab10-s.asm readme-1.txt
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ cat name.txt
Меня зовут Сорокин Кирилл
kvsorokin@dk3n60 ~/work/arch-pc/lab10 $ █

```

Рис. 4.10: Выполнение самостоятельной работы

## 5 Выводы

Мы научились писать программы на языке ассемблера, которые взаимодействуют с файлами.

## Список литературы

1. GDB: The GNU Project Debugger. — URL: <https://www.gnu.org/software/gdb/>.
2. GNU Bash Manual. — 2016. — URL: <https://www.gnu.org/software/bash/manual/>.
3. Midnight Commander Development Center. — 2021. — URL: <https://midnightcommander.org/>.
4. NASM Assembly Language Tutorials. — 2021. — URL: <https://asmtutor.com/>.
5. Newham C. Learning the bash Shell: Unix Shell Programming. — O'Reilly Media, 2005. — 354 с. — (In a Nutshell). — ISBN 0596009658. — URL: <http://www.amazon.com/Learning-bash-Shell-Programming-Nutshell/dp/0596009658>.
6. Robbins A. Bash Pocket Reference. — O'Reilly Media, 2016. — 156 с. — ISBN 978-1491941591.
7. The NASM documentation. — 2021. — URL: <https://www.nasm.us/docs.php>.
8. Zarrelli G. Mastering Bash. — Packt Publishing, 2017. — 502 с. — ISBN 9781784396879.
9. Колдаев В. Д., Лупин С. А. Архитектура ЭВМ. — М. : Форум, 2018.
10. Куляс О. Л., Никитин К. А. Курс программирования на ASSEMBLER. — М. : Солон-Пресс, 2017.
11. Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
12. Расширенный ассемблер: NASM. — 2021. — URL: <https://www.opennet.ru/docs/RUS/nasm/>.
13. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВ- Петербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.
14. Столяров А. Программирование на языке ассемблера NASM для ОС Unix. — 2-е изд. — М. : МАКС Пресс, 2011. — URL: [http://www.stolyarov.info/books/asm\\_unix](http://www.stolyarov.info/books/asm_unix).
15. Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. -

874 с. — (Классика Computer Science).

16. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. -СПб.  
: Питер,

17. — 1120 с. — (Классика Computer Science)