

РУДН. Архитектура компьютеров

Отчёт по лабораторной работе №10

Косинов Никита Андреевич, НПИМбв-02-20

Содержание

1	Цель работы	5
2	Ход работы	6
3	Изменение прав доступа	7
4	Написание программы с записью в файл	13
5	Выводы	18

Список иллюстраций

3.1	Создание файла	7
3.2	Код программы lab10-1	8
3.3	Исполнение lab10-1	10
3.4	Результат работы lab10-1	10
3.5	Попытка исполнения файла, для которого запрещено исполнение	11
3.6	Попытка исполнения текстового файла	11
3.7	Изменение прав доступа в символьном виде	12
3.8	Изменение прав доступа в числовом виде	12
4.1	Основной код программы sr.asm	14
4.2	Подпрограмма _fileprint	16
4.3	СИсполнение и результат sr	17

Список таблиц

1 Цель работы

Очень часто программе необходимо взаимодействовать с файлами: записывать туда данные для дальнейшей работы, брать данные. Цель данной работы - разобраться, как можно создавать, открывать, записывать информацию в файле программой на языке ассемблер, а также разобраться, что такое права доступа и как их изменять.

2 Ход работы

Лабораторная работа выполнена с использованием консоли **ОС Linux** и языка программирования ассемблера **NASM**.

1. Изменение прав доступа;
2. Написание программы с записью в файл.

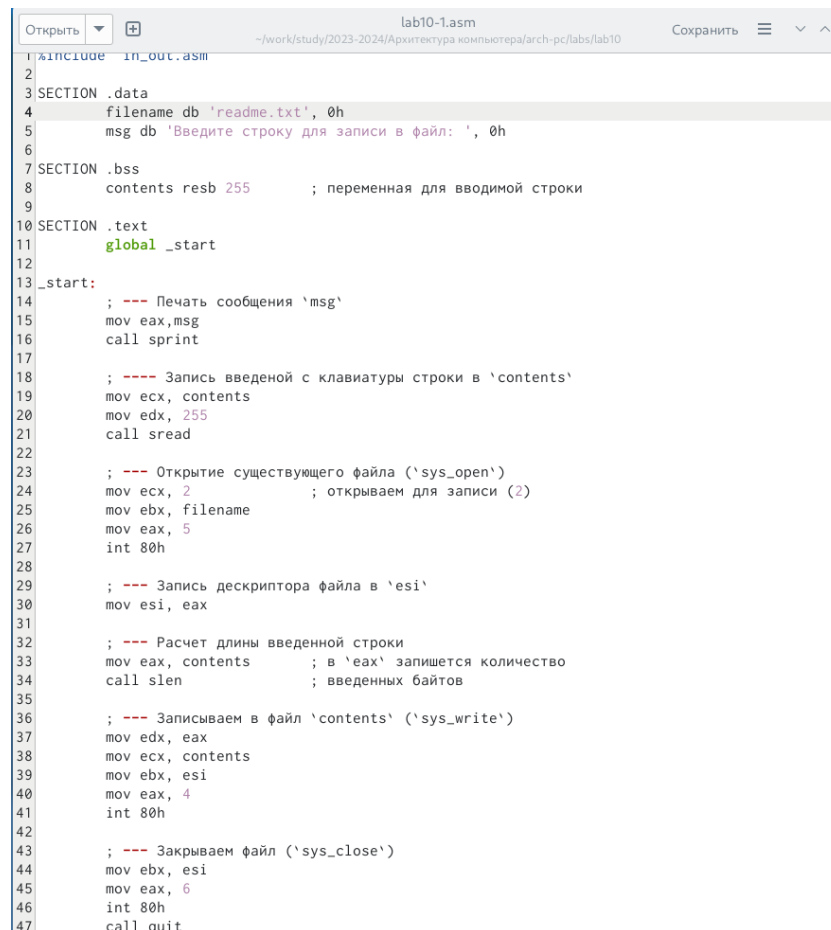
3 Изменение прав доступа

1. Создаём рабочий файл.

```
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc $ cd labs/lab10/  
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ touch lab10-1.asm read  
me-1.txt readme-2.txt
```

Рис. 3.1: Создание файла

2. Пишем код программы, считывающей строку с терминала и записывающую эту строку в файл.



```
1 %include 'in_out.asm'
2
3 SECTION .data
4     filename db 'readme.txt', 0h
5     msg db 'Введите строку для записи в файл: ', 0h
6
7 SECTION .bss
8     contents resb 255 ; переменная для вводимой строки
9
10 SECTION .text
11     global _start
12
13 _start:
14     ; --- Печать сообщения 'msg'
15     mov eax, msg
16     call sprint
17
18     ; --- Запись введенной с клавиатуры строки в 'contents'
19     mov ecx, contents
20     mov edx, 255
21     call sread
22
23     ; --- Открытие существующего файла ('sys_open')
24     mov ecx, 2 ; открываем для записи (2)
25     mov ebx, filename
26     mov eax, 5
27     int 80h
28
29     ; --- Запись дескриптора файла в 'esi'
30     mov esi, eax
31
32     ; --- Расчет длины введенной строки
33     mov eax, contents ; в 'eax' запишется количество
34     call slen ; введенных байтов
35
36     ; --- Записываем в файл 'contents' ('sys_write')
37     mov edx, eax
38     mov ecx, contents
39     mov ebx, esi
40     mov eax, 4
41     int 80h
42
43     ; --- Закрываем файл ('sys_close')
44     mov ebx, esi
45     mov eax, 6
46     int 80h
47     call quit
```

Рис. 3.2: Код программы lab10-1

```
%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 ; 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

```

3. Компилируем и исполняем. Видим, что первая попытка запуска неудачная, ведь в коде открывается уже существующий файл. Поэтому для проверки работы мы создадим файл *readme* и запустим ещё раз.

```

nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ cp ../lab09/in_out.asm
in_out.asm
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ nasm -f elf lab10-1.asm
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ld -m elf_i386 -o lab10-1 lab10-1.o
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ls
in_out.asm lab10-1 lab10-1.asm lab10-1.o presentation readme-1.txt readme-2.txt report
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ./lab10-1
Введите строку для записи в файл: Я загадал число 1, 2 или 3.\n Мне можно задать один вопрос, на который
я могу ответить "Да", "Нет" или "Не знаю". \n Как гарантировано угадать число?
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ды
bash: ды: команда не найдена
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ls
in_out.asm lab10-1 lab10-1.asm lab10-1.o presentation readme-1.txt readme-2.txt report
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ touch readme.txt
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ./lab10-1
Введите строку для записи в файл: Я загадал число 1, 2 или 3.\n Мне можно задать один вопрос, на который
я могу ответить "Да", "Нет" или "Не знаю". \n Как гарантировано угадать число?
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ls
in_out.asm lab10-1 lab10-1.asm lab10-1.o presentation readme-1.txt readme-2.txt readme.txt report
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ gedit readme.txt

```

Рис. 3.3: Исполнение lab10-1

4. Откроем текстовый файл *readme* и убедимся, что программа работает корректно.

```

Открыть  readme.txt  Сохранить
~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10
1 Я загадал число 1, 2 или 3.\n Мне можно задать один вопрос, на который я могу ответить "Да",
"Нет" или "Не знаю". \n Как гарантировано угадать число?

```

Рис. 3.4: Результат работы lab10-1

5. Изменим права доступа командой *chmod*, запретив исполнения для всех категорий пользователей. Проверим корректность командой *ls* с ключом *-l*. Видим, что у файла *lab10-1* в строке прав доступа отсутствуют буквы *x*. Ожидаемо, исполнить файл у нас не выходит.

```
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ chmod ugo-x lab10-1
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ls -l lab10-1
-rw-r--r-- 1 nakosinov studsci 9164 окт 27 16:01 lab10-1
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ./lab10-1
bash: ./lab10-1: Отказано в доступе
```

Рис. 3.5: Попытка исполнения файла, для которого запрещено исполнение

6. Добавим право запускать файл владельцу файла *lab10-1.asm*. Попробуем исполнить его. Но этот файл - не скомпилированная программа, и ЭВМ не понимает, как он может его исполнить, показывая нам, что не знает команд, введенных в этот файл.

```
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ls -l lab10-1.asm
-rw-r--r-- 1 nakosinov studsci 1155 окт 27 15:59 lab10-1.asm
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ chmod u+x lab10-1.asm
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ls -l lab10-1.asm
-rwxr--r-- 1 nakosinov studsci 1155 окт 27 15:59 lab10-1.asm
nakosinov@dk8n52 ~/work/study/2023-2024/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 8: переменная: 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: синтаксическая ошибка рядом с неожиданным маркером '<';>'
./lab10-1.asm: строка 14: '      ; --- Печать сообщения 'msg'
```

Рис. 3.6: Попытка исполнения текстового файла

7. Изменим права доступа к файлу *readme-1* в символьном виде в соответствии с вариантом 1: *-x -wx rwx*. В ходе нескольких попыток видим, что задать сразу все права таким образом не очень удобно: приходится вводить несколько команд.

```

nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ chmod ugo---x-wxrwrx readme-1.txt
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ls -l readme-1.txt
-----1 nakosinov studsci 0 окт 27 15:57 readme-1.txt
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ chmod u---x readme-1.txt
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ chmod g=wx readme-1.txt
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ chmod o=rwx readme-1.txt
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ls -l readme-1.txt
-----rwx 1 nakosinov studsci 0 окт 27 15:57 readme-1.txt
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ chmod g=wx readme-1.txt
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ls -l readme-1.txt
-----rwx 1 nakosinov studsci 0 окт 27 15:57 readme-1.txt
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ chmod u+x readme-1.txt
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ chmod g+wx readme-1.txt
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ls -l readme-1.txt
---x-wxrwrx 1 nakosinov studsci 0 окт 27 15:57 readme-1.txt

```

Рис. 3.7: Изменение прав доступа в символьном виде

8. Изменим права доступа к файлу *readme-2* в двоичном виде в соответствии с вариантом 1: *000 110 010*. Для этого переводим каждую тройку бит в десятичную цифру. Одной командой настраиваем права доступа полностью.

```

nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ chmod 062 readme-2.txt
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ls -l readme-2.txt
----rw--w- 1 nakosinov studsci 0 окт 27 15:57 readme-2.txt

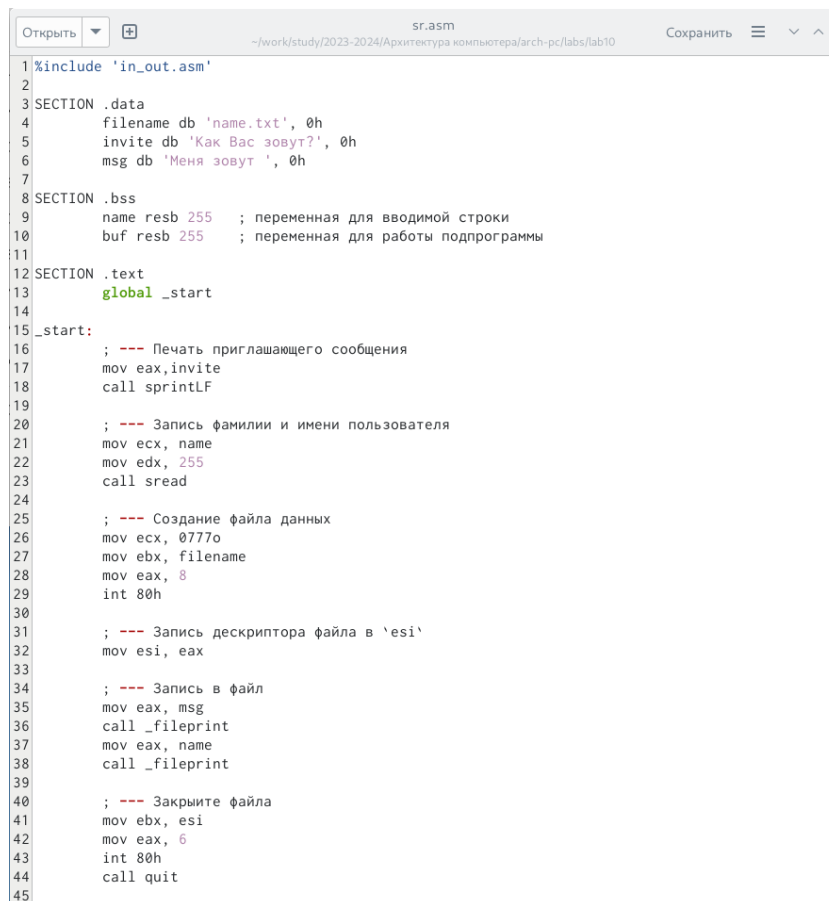
```

Рис. 3.8: Изменение прав доступа в числовом виде

4 Написание программы с записью в файл

Задание для самостоятельной работы заключается в написании программы, запрашивающей имя пользователя и записывающей это имя в файл, добавляя в начало приветственные слова.

1. Создаём файл с кодом программы и добавляем комментарии.



```
1 %include 'in_out.asm'
2
3 SECTION .data
4     filename db 'name.txt', 0h
5     invite db 'Как Вас зовут?', 0h
6     msg db 'Меня зовут ', 0h
7
8 SECTION .bss
9     name resb 255 ; переменная для вводимой строки
10    buf resb 255 ; переменная для работы подпрограммы
11
12 SECTION .text
13     global _start
14
15 _start:
16     ; --- Печать приглашающего сообщения
17     mov eax, invite
18     call sprintf
19
20     ; --- Запись фамилии и имени пользователя
21     mov ecx, name
22     mov edx, 255
23     call sread
24
25     ; --- Создание файла данных
26     mov ecx, 0777o
27     mov ebx, filename
28     mov eax, 8
29     int 80h
30
31     ; --- Запись дескриптора файла в 'esi'
32     mov esi, eax
33
34     ; --- Запись в файл
35     mov eax, msg
36     call _fileprint
37     mov eax, name
38     call _fileprint
39
40     ; --- Закройте файла
41     mov ebx, esi
42     mov eax, 6
43     int 80h
44     call quit
45
```

Рис. 4.1: Основной код программы sr.asm

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

```
SECTION .data
```

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

```
    invite db 'Как Вас зовут?', 0h
```

```
    msg db 'Меня зовут ', 0h
```

```
SECTION .bss
```

```
    name resb 255 ; переменная для вводимой строки
```

```
    buf resb 255 ; переменная для работы подпрограммы
```

```

SECTION .text

    global _start

_start:

    ; --- Печать приглашающего сообщения
    mov eax, invite
    call sprintfLF

    ; --- Запись фамилии и имени пользователя
    mov ecx, name
    mov edx, 255
    call sread

    ; --- Создание файла данных
    mov ecx, 0777o
    mov ebx, filename
    mov eax, 8
    int 80h

    ; --- Запись дескриптора файла в `esi`
    mov esi, eax

    ; --- Запись в файл
    mov eax, msg
    call _fileprint
    mov eax, name
    call _fileprint

    ; --- Закройте файла

```

```

mov ebx, esi
mov eax, 6
int 80h
call quit

; --- Подпрограмма записи в файл
_fileprint:
    mov ecx, eax
    call slen
    mov edx, eax
    mov ebx, esi
    mov eax, 4
    int 80h
    ret

```

2. В файл необходимо дважды записать сообщения: приветственное “*Меня зовут*” и непосредственно имя пользователя. Поэтому, для удобства, выносим печать в файл в подпрограмму *_fileprint*.

```

45
46 ; --- Подпрограмма записи в файл
47 _fileprint:
48     mov ecx, eax
49     call slen
50     mov edx, eax
51     mov ebx, esi
52     mov eax, 4
53     int 80h
54     ret

```

Рис. 4.2: Подпрограмма _fileprint

3. Компилируем программу и запускаем. Вводим свои имя и фамилию. Проверяем, что файл с именем создан командой *ls*. Командой *cat* проверяем, что в созданном файле данные сохранены и в нужном нам формате.


```

nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ gedit sr.asm
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ nasm -f elf sr.asm
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ld -m elf_i386 -o sr s
r.o
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ./sr
Как Вас зовут?
Никита Косинов
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ ls
in_out.asm  lab10-1.asm  name.txt    readme-1.txt  readme.txt  sr          sr.o
lab10-1     lab10-1.o    presentation  readme-2.txt  report      sr.asm
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $ cat name.txt
Меня зовут Никита Косинов
nakosinov@dk8n52 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab10 $

```

Рис. 4.3: СИсполнение и результат sr

5 Выводы

В ходе данной лабораторной работы мы научились работать с файлами средствами **NASM**, а также добавлять и отнимать права доступа к файлам разным группам пользователей через терминал.