

# **Отчёт по лабораторной работе №10**

**Дисциплина: Архитектура компьютера**

Калашникова Ольга Сергеевна

# Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выполнение заданий для самостоятельной работы	12
4	Выводы	17

## Список иллюстраций

2.1	Создание папки, её открытие и создание файла . . . . .	6
2.2	Проверка наличия папки и файлов . . . . .	6
2.3	Текст программы . . . . .	7
2.4	Перемещение файла “in_out.asm” . . . . .	8
2.5	Создание и запуск исполняемого файла . . . . .	8
2.6	Проверка файла readme.txt . . . . .	8
2.7	Изменение прав доступа к исполняемому файлу и попытка выпол- нить файл . . . . .	9
2.8	Изменение прав доступа к файлу и попытка выполнить файл . . .	9
2.9	Изменение прав доступа к файлу readme-1.txt . . . . .	10
2.10	Изменение прав доступа к файлу readme-2.txt . . . . .	10
2.11	Таблица . . . . .	11
3.1	Создание файла . . . . .	12
3.2	Проверка наличия файла . . . . .	13
3.3	Написание программы . . . . .	14
3.4	Создание и запуск исполняемого файла . . . . .	14
3.5	Проверка наличия файла и его содержимого . . . . .	15

## **Список таблиц**

# 1 Цель работы

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

## 2 Выполнение лабораторной работы

1. Создаём каталог для программ лабораторной работы № 10(при помощи команды `mkdir ~/work/arch-pc/lab10`), переходим в него (при помощи команды `cd ~/work/arch-pc/lab10`) и создаём файлы `lab10-1.asm`, `readme-1.txt`, `readme-2.txt` и `readme.txt` (при помощи команды `touch lab10-1.asm readme-1.txt readme-2.txt readme.txt`) (рис. 2.1),(рис. 2.2)

```
oskashnikova@dk6n62:~$ mkdir ~/work/arch-pc/lab10
oskashnikova@dk6n62:~$ cd ~/work/arch-pc/lab10
oskashnikova@dk6n62:~/work/arch-pc/lab10$ touch lab10-1.asm readme-1.txt readme-2.txt readme.txt
oskashnikova@dk6n62:~/work/arch-pc/lab10$ █
```

Рис. 2.1: Создание папки, её открытие и создание файла

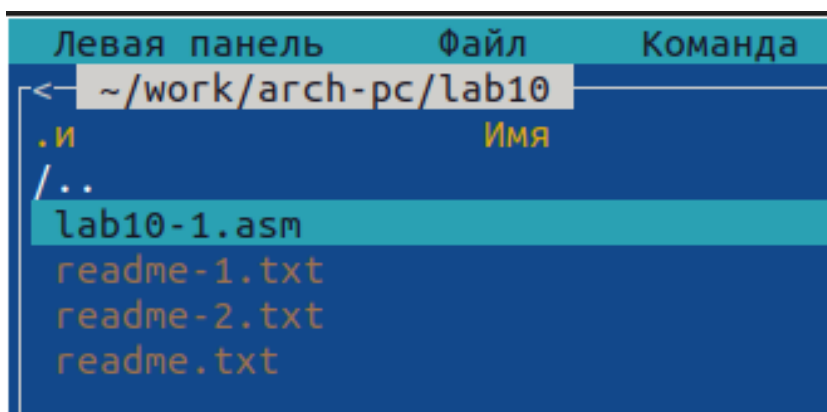
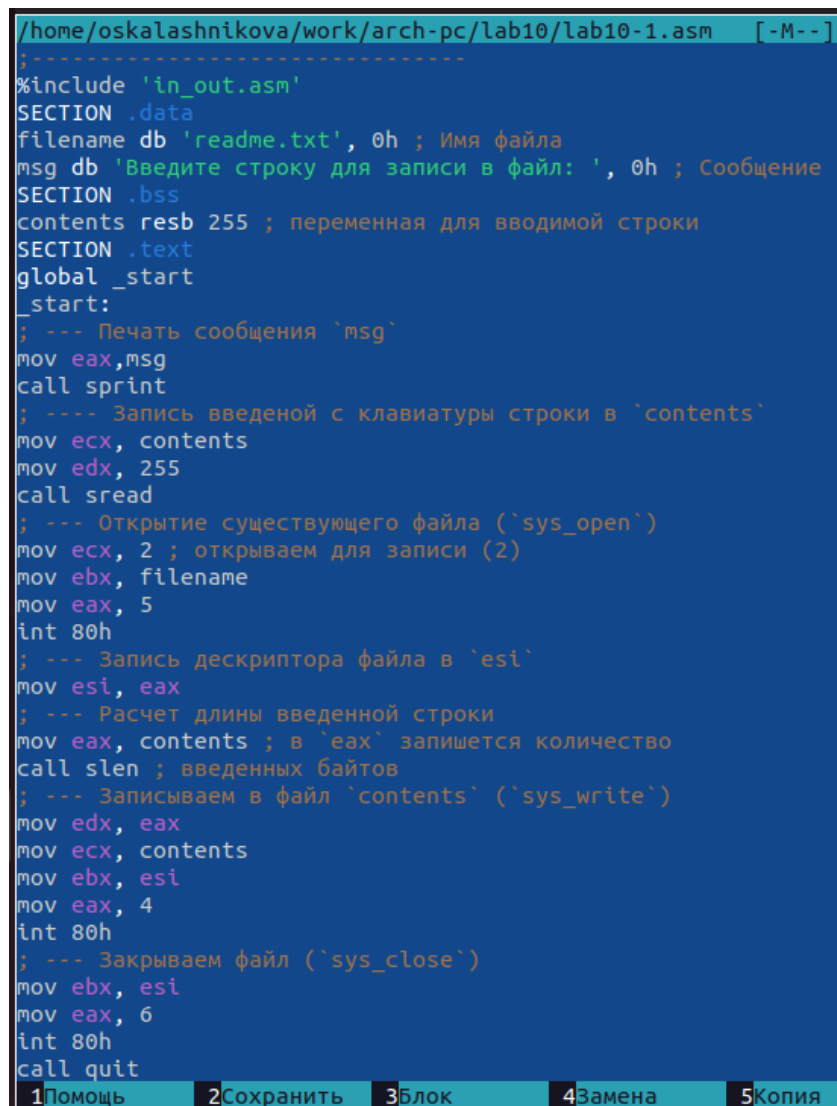


Рис. 2.2: Проверка наличия папки и файлов

2. Введём в файл `lab10-1.asm` текст программы из листинга 10.1 (Программа записи в файл сообщения). Для корректной работы нужно переместить файл

“in\_out.asm” в тот же каталог, где лежит и файл с текстом программы.(рис. 2.3),(рис. 2.4)



```
/home/oskalashnikova/work/arch-pc/lab10/lab10-1.asm [-M--]
;-----
%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
```

1Помощь 2Сохранить 3Блок 4Замена 5Копия

Рис. 2.3: Текст программы

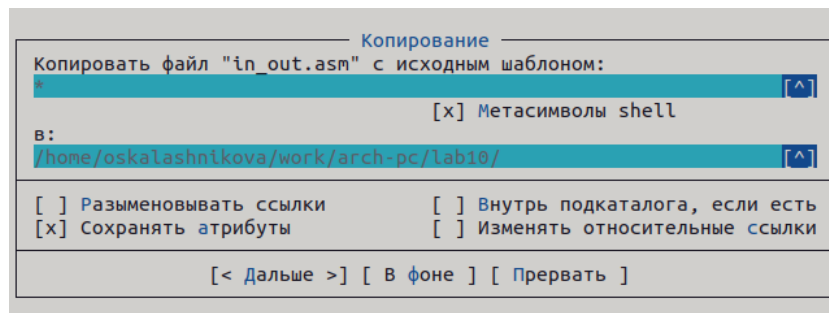


Рис. 2.4: Перемещение файла “in\_out.asm”

Создаем исполняемый файл и запускаем его (компиляция: `nasm -f elf lab10-1.asm`, `ld -m elf_i386 -o lab10-1 lab10-1.o`, запуск: `./lab10-1`). Для записи используем `readme.txt` и после проверяем записался ли текст введенный с клавиатуры (рис. 2.5),(рис. 2.6)

```
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ nasm -f elf lab10-1.asm
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ ld -m elf_i386 -o lab10-1 lab10-1.o
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ ./lab10-1
Введите строку для записи в файл: Hello world!
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ ls -l
итого 28
-rw-rw-r-- 1 oskalashnikova oskalashnikova 3942 ноя  5 10:34 in_out.asm
-rwxrwxr-x 1 oskalashnikova oskalashnikova 9164 дек 16 14:41 lab10-1
-rw-rw-r-- 1 oskalashnikova oskalashnikova 1286 дек 16 14:22 lab10-1.asm
-rw-rw-r-- 1 oskalashnikova oskalashnikova 1472 дек 16 14:40 lab10-1.o
-rw-rw-r-- 1 oskalashnikova oskalashnikova    0 дек 16 14:16 readme-1.txt
-rw-rw-r-- 1 oskalashnikova oskalashnikova    0 дек 16 14:16 readme-2.txt
-rw-rw-r-- 1 oskalashnikova oskalashnikova   13 дек 16 14:41 readme.txt
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ cat readme.txt
Hello world!
oskalashnikova@dk6n62:~/work/arch-pc/lab10$
```

Рис. 2.5: Создание и запуск исполняемого файла

```
/home/oskalashnikova/work/arch-pc/lab10/readme.txt
Hello World!
```

Рис. 2.6: Проверка файла readme.txt

3. С помощью команды `chmod u-x lab10-1` изменим права доступа к исполняемому файлу `lab10-1`, запретив его выполнение. Попытаемся выполнить файл (рис. 2.7)



```
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ chmod u-x lab10-1
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ ./lab10-1
bash: ./lab10-1: Отказано в доступе
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ ls -l
итого 28
-rw-rw-r-- 1 oskalashnikova oskalashnikova 3942 ноя  5 10:34 in_out.asm
-rw-rwxr-x 1 oskalashnikova oskalashnikova 9164 дек 16 16:23 lab10-1
-rw-rw-r-- 1 oskalashnikova oskalashnikova 1286 дек 16 16:16 lab10-1.asm
-rw-rw-r-- 1 oskalashnikova oskalashnikova 1472 дек 16 16:22 lab10-1.o
-rw-rw-r-- 1 oskalashnikova oskalashnikova   0 дек 16 16:15 readme-1.txt
-rw-rw-r-- 1 oskalashnikova oskalashnikova   0 дек 16 16:15 readme-2.txt
-rw-rw-r-- 1 oskalashnikova oskalashnikova  13 дек 16 16:23 readme.txt
oskalashnikova@dk6n62:~/work/arch-pc/lab10$
```

Рис. 2.7: Изменение прав доступа к исполняемому файлу и попытка выполнить файл

Эта команда убирает право (- Отменить набор прав) на выполнение (x Право на исполнение) у пользователя (u это Владелец, то есть я) для файла "lab10-1". Теперь, если попробуем выполнить этот файл, мы получим сообщение об ошибке, потому что у нас больше нет прав на его выполнение.

4. Изменим права доступа к файлу с исходным текстом программы, добавив права на исполнение при помощи команды `chmod u+x lab10-1.asm`. Пробуем запустить файл (рис. 2.8)

```
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ chmod u+x lab10-1.asm
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ ./lab10-1.asm
./lab10-1.asm: строка 1: синтаксическая ошибка рядом с неожиданным маркером «;»
./lab10-1.asm: строка 1: `;-----'
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ ls -l
итого 28
-rw-rw-r-- 1 oskalashnikova oskalashnikova 3942 ноя  5 10:34 in_out.asm
-rw-rwxr-x 1 oskalashnikova oskalashnikova 9164 дек 16 16:23 lab10-1
-rwxrwxr-- 1 oskalashnikova oskalashnikova 1286 дек 16 16:16 lab10-1.asm
-rw-rw-r-- 1 oskalashnikova oskalashnikova 1472 дек 16 16:22 lab10-1.o
-rw-rw-r-- 1 oskalashnikova oskalashnikova   0 дек 16 16:15 readme-1.txt
-rw-rw-r-- 1 oskalashnikova oskalashnikova   0 дек 16 16:15 readme-2.txt
-rw-rw-r-- 1 oskalashnikova oskalashnikova  13 дек 16 16:23 readme.txt
oskalashnikova@dk6n62:~/work/arch-pc/lab10$
```

Рис. 2.8: Изменение прав доступа к файлу и попытка выполнить файл

Эта команда добавляет право (+ Добавить набор прав) на выполнение (x Право на исполнение) у пользователя (u это Владелец, то есть я) для файла "lab10-1.asm". lab10-1.asm является файлом с исходным кодом программы на языке ассемблера и искусственно добавленные права на исполнение не дадут возможности

выполнить его напрямую, как исполняемый файл. Для того чтобы этот файл выполнялся, необходимо сначала скомпилировать его в исполняемый файл.

5. В соответствии с вариантом в таблице 10.4 предоставим права доступа к файлу `readme-1.txt` представленные в символьном виде, а для файла `readme-2.txt` – в двоичном виде. Проверим правильность выполнения с помощью команды `ls -l`. Номер моего варианта - 7. (рис. 2.9), (рис. 2.10).

```
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ chmod 676 readme-1.txt
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ ls -l
итого 28
-rw-rw-r-- 1 oskalashnikova oskalashnikova 3942 ноя  5 10:34 in_out.asm
-rw-rwxr-x 1 oskalashnikova oskalashnikova 9164 дек 16 16:23 lab10-1
-rwxrw-r-- 1 oskalashnikova oskalashnikova 1286 дек 16 16:16 lab10-1.asm
-rw-rw-r-- 1 oskalashnikova oskalashnikova 1472 дек 16 16:22 lab10-1.o
-rw-rwxrw- 1 oskalashnikova oskalashnikova   0 дек 16 16:15 readme-1.txt
-rw-rw-r-- 1 oskalashnikova oskalashnikova   0 дек 16 16:15 readme-2.txt
-rw-rw-r-- 1 oskalashnikova oskalashnikova  13 дек 16 16:23 readme.txt
oskalashnikova@dk6n62:~/work/arch-pc/lab10$
```

Рис. 2.9: Изменение прав доступа к файлу `readme-1.txt`

```
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ chmod 577 readme-2.txt
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ ls -l
итого 28
-rw-rw-r-- 1 oskalashnikova oskalashnikova 3942 ноя  5 10:34 in_out.asm
-rw-rwxr-x 1 oskalashnikova oskalashnikova 9164 дек 16 16:23 lab10-1
-rwxrw-r-- 1 oskalashnikova oskalashnikova 1286 дек 16 16:16 lab10-1.asm
-rw-rw-r-- 1 oskalashnikova oskalashnikova 1472 дек 16 16:22 lab10-1.o
-rw-rwxrw- 1 oskalashnikova oskalashnikova   0 дек 16 16:15 readme-1.txt
-r-xrw-rwx 1 oskalashnikova oskalashnikova   0 дек 16 16:15 readme-2.txt
-rw-rw-r-- 1 oskalashnikova oskalashnikova  13 дек 16 16:23 readme.txt
oskalashnikova@dk6n62:~/work/arch-pc/lab10$
```

Рис. 2.10: Изменение прав доступа к файлу `readme-2.txt`

7 вариант: `rw- rwx rw- 101 111 111`. Смотрим значения по таблице в которой указаны двоичный, буквенный и восмеричный способ записи триады прав доступа (рис. 2.11).

**Таблица 10.1.** Двоичный, буквенный и восьмеричный способ записи триады прав доступа

Двоичный	Буквенный	Восьмеричный
111	rwx	7
110	rw-	6
101	r-x	5
100	r--	4
011	-wx	3
010	-w-	2
001	--x	1
000	---	0

Рис. 2.11: Таблица

### 3 Выполнение заданий для самостоятельной работы

Создаём файл lab10-2.asm с помощью touch lab10-2.asm (рис. 3.1), (рис. 3.2).

```
oskashnikova@dk6n62:~/work/arch-pc/lab10$ touch lab10-2.asm  
oskashnikova@dk6n62:~/work/arch-pc/lab10$ █
```

Рис. 3.1: Создание файла

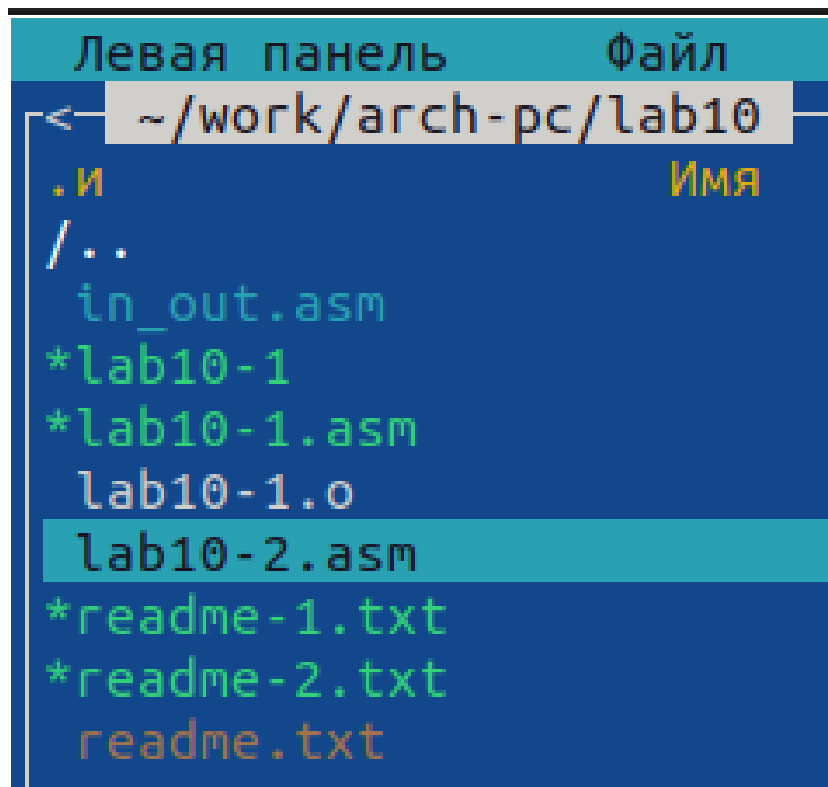


Рис. 3.2: Проверка наличия файла

Вводим в созданный файл текст программы, которая выполнит представленный список действий (рис. 3.3).

```

/home/oskalashnikova/work/arch-pc/lab10/lab10-2.asm
#include 'in_out.asm'

SECTION .data
msg: DB 'Как вас зовут? ',0
file: DB 'name.txt',0
name: DB 'Меня зовут ',0

SECTION .bss
X: RESB 80

SECTION .text
GLOBAL _start
_start:
;Вывод приглашения "Как Вас зовут?"
    mov eax,msg
    call sprint
;ввести с клавиатуры свои фамилию и имя
    mov ecx,X
    mov edx,80
    call sread
;создать файл с именем name.txt
    mov ecx,0777o
    mov ebx,file
    mov eax,8
    int 80h
;записать в файл сообщение "Меня зовут"
    mov ecx, 2
    mov ebx, file
    mov eax, 5
    int 80h
    mov esi, eax
    mov eax, name
    call slen
    mov edx, eax
    mov ecx, name
    mov ebx, esi
    mov eax, 4
    int 80h

```

1Помощь 2Сохранить 3Блок 4Замена

Рис. 3.3: Написание программы

Создаем исполняемый файл и запускаем его (рис. 3.4)

```

oskalashnikova@dk6n62:~/work/arch-pc/lab10$ nasm -f elf lab10-2.asm
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ ld -m elf_i386 -o lab10-2 lab10-2.o
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ ./lab10-2
Как вас зовут? Olga kalashnikova

```

Рис. 3.4: Создание и запуск исполняемого файла

Проверим наличие файла и его содержимое с помощью команд ls и cat (рис. 3.5)

```
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ ls
in_out.asm lab10-1 lab10-1.asm lab10-1.o lab10-2 lab10-2.asm lab10-2.o name.txt readme-1.txt readme-2.txt readme.txt
oskalashnikova@dk6n62:~/work/arch-pc/lab10$ cat name.txt
Меня зовут Olga kalashnikova
oskalashnikova@dk6n62:~/work/arch-pc/lab10$
```

Рис. 3.5: Проверка наличия файла и его содержимого

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

```
SECTION .data
```

```
msg: DB 'Как вас зовут? ',0
```

```
file: DB 'name.txt',0
```

```
name: DB 'Меня зовут ',0
```

```
SECTION .bss
```

```
X: RESB 80
```

```
SECTION .text
```

```
GLOBAL _start
```

```
_start:
```

```
;Вывод приглашения "Как Вас зовут?"
```

```
    mov eax,msg
```

```
    call sprint
```

```
;ввести с клавиатуры свои фамилию и имя
```

```
    mov ecx,X
```

```
    mov edx,80
```

```
    call sread
```

```
;создать файл с именем name.txt
```

```
    mov ecx,0777o
```

```
    mov ebx,file
```

```

    mov eax,8
    int 80h
;записать в файл сообщение "Меня зовут"
    mov ecx, 2
    mov ebx, file
    mov eax, 5
    int 80h
    mov esi, eax
    mov eax, name
    call slen
    mov edx, eax
    mov ecx, name
    mov ebx, esi
    mov eax, 4
    int 80h
;дописать в файл строку введенную с клавиатуры
    mov eax,X
    call slen
    mov edx,eax
    mov ecx,X
    mov ebx,esi
    mov eax,4
    int 80h
;закрыть файл
    mov ebx,esi
    mov eax,6
    int 80h
    call quit

```



## 4 Выводы

В ходе лабораторной работы мы научились писать программы для работы с файлам и научились предоставлять права доступа к файлам.