

Отчёт по лабораторной работе 10

Дисциплина: архитектура компьютера

Айдарбекова Алия НММбд-01-23

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	9
4.1	Задание для самостоятельной работы	13
5	Выводы	16

Список иллюстраций

4.1	Изменение кода lab10-1.asm	10
4.2	Компиляция текста программы lab10-1.asm	11
4.3	файл без возможности запуска	11
4.4	файл asm с разрешением на выполнение	12
4.5	установка прав	13
4.6	Изменение кода lab10-2.asm	14
4.7	Компиляция текста программы lab10-2.asm	15

Список таблиц

1 Цель работы

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

2 Задание

1. Изучение работы с файлами в ассемблере
2. Изучение примеров программ
3. Изучение прав доступа
4. Выполнение заданий для самостоятельной работы

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. Владелцем файла является его создатель.

Набор прав доступа задается тройками битов и состоит из прав на чтение, запись и исполнение файла. В символьном представлении он имеет вид строк `gwx`, где вместо любого символа может стоять дефис. Всего возможно 8 комбинаций, приведенных в таблице 10.1. Буква означает наличие права (установлен в единицу второй бит триады `r` — чтение, первый бит `w` — запись, нулевой бит `x` — исполнение), а дефис означает отсутствие права (нулевое значение соответствующего бита). Также права доступа могут быть представлены как восьмеричное число. Так, права доступа `rw-` (чтение и запись, без исполнения) понимаются как три двоичные цифры `110` или как восьмеричная цифра `6`.

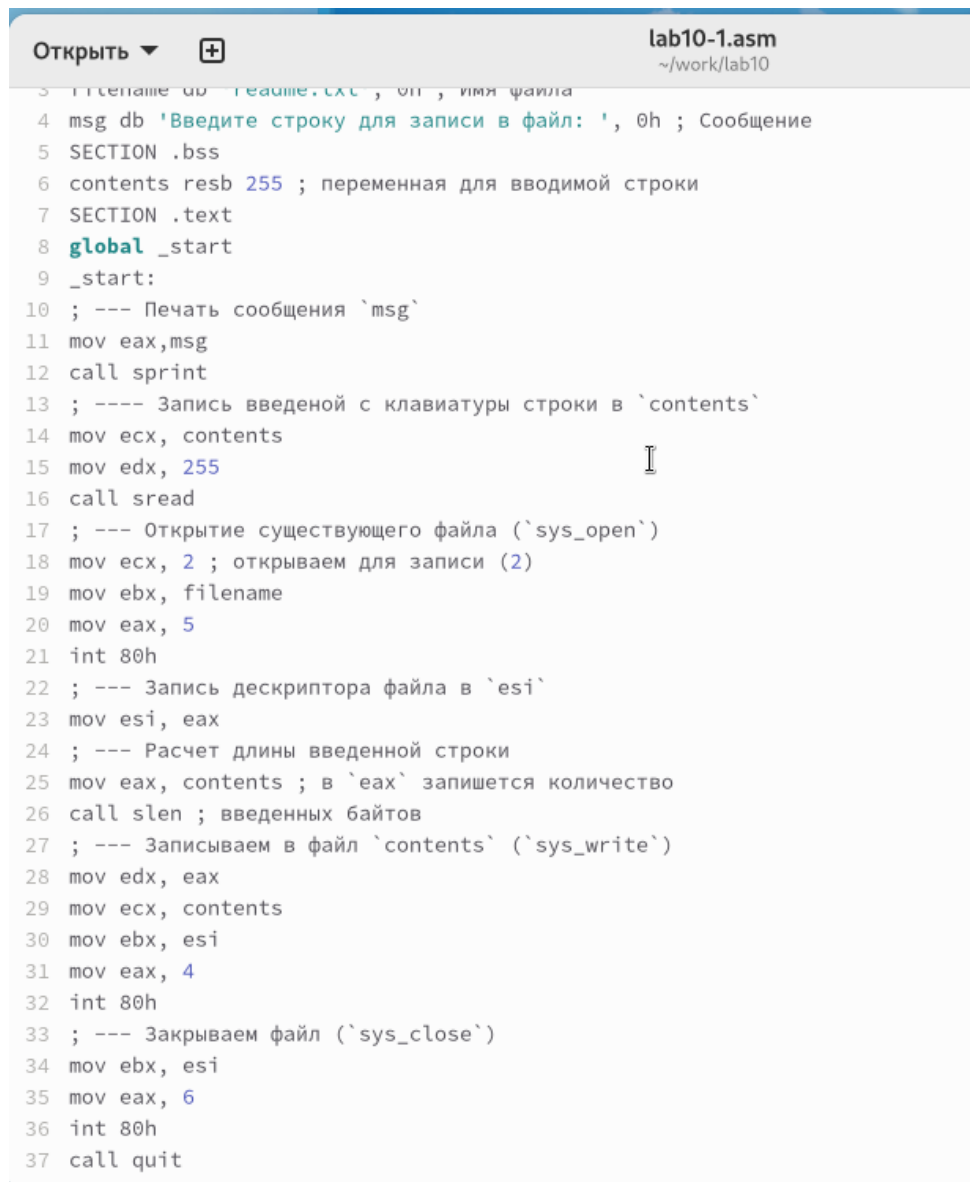
Обработка файлов в операционной системе Linux осуществляется за счет использования определенных системных вызовов. Для корректной работы и доступа к файлу при его открытии или создании, файлу присваивается уникальный

номер (16-битное целое число) – дескриптор файла

4 Выполнение лабораторной работы

Я организовала каталог для лабораторной работы №10 и перешла в него. Внутри каталога, мною были созданы три файла: lab10-1.asm, readme-1.txt и readme-2.txt.

В файле lab10-1.asm, я разработала программу в соответствии с листингом 10.1, которая записывает сообщение в файл. После этого, я преобразовала этот код в исполняемый файл и проверила его функционирование.(рис. [4.1])



```
Открыть ▾ + lab10-1.asm
~/work/lab10
3 filename db 'readme.txt', 0h, имя файла
4 msg db 'Введите строку для записи в файл: ', 0h ; Сообщение
5 SECTION .bss
6 contents resb 255 ; переменная для вводимой строки
7 SECTION .text
8 global _start
9 _start:
10 ; --- Печать сообщения `msg`
11 mov eax,msg
12 call sprint
13 ; ---- Запись введенной с клавиатуры строки в `contents`
14 mov ecx, contents
15 mov edx, 255
16 call sread
17 ; --- Открытие существующего файла (`sys_open`)
18 mov ecx, 2 ; открываем для записи (2)
19 mov ebx, filename
20 mov eax, 5
21 int 80h
22 ; --- Запись дескриптора файла в `esi`
23 mov esi, eax
24 ; --- Расчет длины введенной строки
25 mov eax, contents ; в `eax` запишется количество
26 call slen ; введенных байтов
27 ; --- Записываем в файл `contents` (`sys_write`)
28 mov edx, eax
29 mov ecx, contents
30 mov ebx, esi
31 mov eax, 4
32 int 80h
33 ; --- Закрываем файл (`sys_close`)
34 mov ebx, esi
35 mov eax, 6
36 int 80h
37 call quit
```

Рис. 4.1: Изменение кода lab10-1.asm

Эта программа запрашивает строку и перезаписывает её в файл readme.txt. Однако, если указанный файл не существует, строка не будет сохранена нигде.(рис. [4.2]) (рис. [4.2])

```
[arraydarbekova@fedora lab10]$ nasm -f elf lab10-1.asm
[arrayjdarbekova@fedora lab10]$ ld -m elf_i386 -o lab10-1 lab10-1.o
[arrayjdarbekova@fedora lab10]$ ./lab10-1
Введите строку для записи в файл: Aliya
[arrayjdarbekova@fedora lab10]$ cat readme.txt
Aliya
[arrayjdarbekova@fedora lab10]$
```

Рис. 4.2: Компиляция текста программы lab10-1.asm

Для того чтобы отменить выполнение исполняемого файла lab10-1, я применила команду `chmod` для изменения прав доступа. Я убрала атрибут “x” во всех трех позициях. После этого, я попыталась запустить файл.

Но файл не запускался, так как атрибут “x” был отсутствовал во всех трех позициях, что запрещало его выполнение. (рис. [4.3])

```
[arrayjdarbekova@fedora lab10]$ chmod -x lab10-1
[arrayjdarbekova@fedora lab10]$ ./lab10-1
bash: ./lab10-1: Отказано в доступе
[arrayjdarbekova@fedora lab10]$
```

Рис. 4.3: файл без возможности запуска

Я изменила настройки доступа к файлу lab10-1.asm, добавив разрешение на его выполнение с использованием команды `chmod`. После этого, я попыталась запустить файл. (рис. [4.4])

В итоге, файл был активирован, и терминал попытался интерпретировать его содержание как команды командной строки. Однако, поскольку это файл с кодом на языке ассемблера, а не команды для терминала, произошли ошибки. Но, если в такой файл внести команды командной строки, то их можно будет выполнить, запустив файл.

```

[arayjdarbekova@fedora lab10]$ chmod +x lab10-1.asm
[arayjdarbekova@fedora lab10]$ ./lab10-1.asm
./lab10-1.asm: строка 1: fg: нет управления заданиями
./lab10-1.asm: строка 2: SECTION: команда не найдена
./lab10-1.asm: строка 3: filename: команда не найдена
./lab10-1.asm: строка 3: Имя: команда не найдена
./lab10-1.asm: строка 4: msg: команда не найдена
./lab10-1.asm: строка 4: Сообщение: команда не найдена
./lab10-1.asm: строка 5: SECTION: команда не найдена
./lab10-1.asm: строка 6: contents: команда не найдена
./lab10-1.asm: строка 6: переменная: команда не найдена
./lab10-1.asm: строка 7: SECTION: команда не найдена
./lab10-1.asm: строка 8: global: команда не найдена
./lab10-1.asm: строка 9: _start:: команда не найдена
./lab10-1.asm: строка 10: синтаксическая ошибка рядом с неожиданным маркером «;»
./lab10-1.asm: строка 10: `; --- Печать сообщения `msg`'
[arayjdarbekova@fedora lab10]$

```

Рис. 4.4: файл asm с разрешением на выполнение

Затем, я настроила права доступа к файлам readme в соответствии с указаниями в таблице 10.4. Чтобы убедиться в правильности выполнения, я применила команду `ls -l`. (рис. [4.5])

для варианта 14: `r-x rwx rwx 110 111 110`

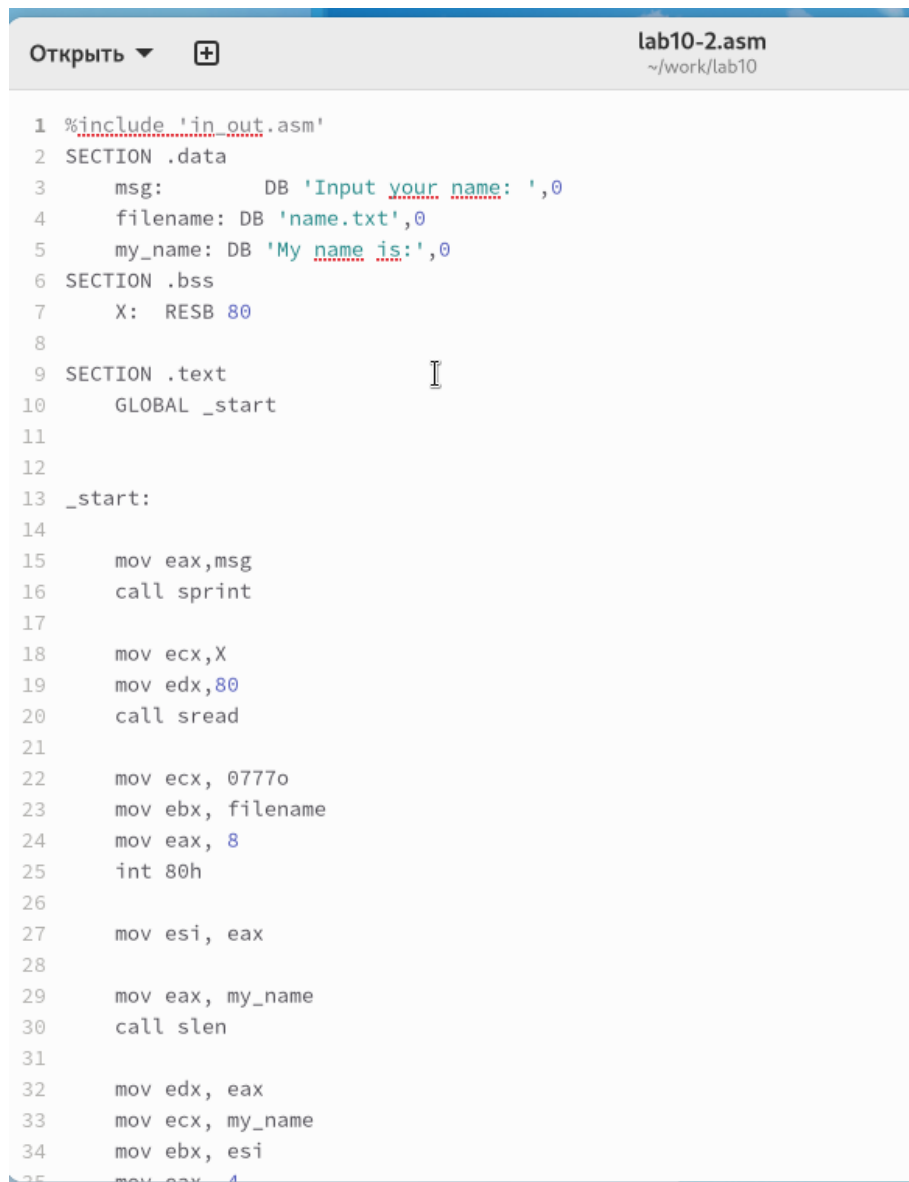
```
[arayjdarbekova@fedora lab10]$ ls -l
итого 28
-rw-----. 1 arayjdarbekova arayjdarbekova 3773 дек 23 10:28 in_out.asm
-rw-r--r--. 1 arayjdarbekova arayjdarbekova 9164 дек 23 10:29 lab10-1
-rwx--x--x. 1 arayjdarbekova arayjdarbekova 1140 дек 23 10:28 lab10-1.asm
-rw-r--r--. 1 arayjdarbekova arayjdarbekova 1472 дек 23 10:29 lab10-1.o
-rw-----. 1 arayjdarbekova arayjdarbekova    0 дек 23 10:28 readme-2.txt
-rw-----. 1 arayjdarbekova arayjdarbekova    6 дек 23 10:30 readme.txt
[arayjdarbekova@fedora lab10]$
[arayjdarbekova@fedora lab10]$ chmod 577 readme.txt
[arayjdarbekova@fedora lab10]$ chmod 676 readme-2.txt
[arayjdarbekova@fedora lab10]$ ls -l
итого 28
-rw-----. 1 arayjdarbekova arayjdarbekova 3773 дек 23 10:28 in_out.asm
-rw-r--r--. 1 arayjdarbekova arayjdarbekova 9164 дек 23 10:29 lab10-1
-rwx--x--x. 1 arayjdarbekova arayjdarbekova 1140 дек 23 10:28 lab10-1.asm
-rw-r--r--. 1 arayjdarbekova arayjdarbekova 1472 дек 23 10:29 lab10-1.o
-rw-rwxrw-. 1 arayjdarbekova arayjdarbekova    0 дек 23 10:28 readme-2.txt
-r-xrwxrwx. 1 arayjdarbekova arayjdarbekova    6 дек 23 10:30 readme.txt
[arayjdarbekova@fedora lab10]$
```

Рис. 4.5: установка прав

4.1 Задание для самостоятельной работы

Написала программу работающую по следующему алгоритму (рис. [4.6]) (рис. [4.7]):

- Вывод приглашения “Как Вас зовут?”
- ввести с клавиатуры свои фамилию и имя
- создать файл с именем name.txt
- записать в файл сообщение “Меня зовут”
- дописать в файл строку введенную с клавиатуры
- закрыть файл



```
1 %include 'in_out.asm'
2 SECTION .data
3     msg:      DB 'Input your name: ',0
4     filename: DB 'name.txt',0
5     my_name:  DB 'My name is:',0
6 SECTION .bss
7     X:  RESB 80
8
9 SECTION .text
10     GLOBAL _start
11
12
13 _start:
14
15     mov eax,msg
16     call sprint
17
18     mov ecx,X
19     mov edx,80
20     call sread
21
22     mov ecx, 0777o
23     mov ebx, filename
24     mov eax, 8
25     int 80h
26
27     mov esi, eax
28
29     mov eax, my_name
30     call slen
31
32     mov edx, eax
33     mov ecx, my_name
34     mov ebx, esi
35     mov eax, 4
```

Рис. 4.6: Изменение кода lab10-2.asm

```

[arayjdarbekova@fedora lab10]$ nasm -f elf lab10-2.asm
[arayjdarbekova@fedora lab10]$ ld -m elf_i386 -o lab10-2 lab10-2.o
[arayjdarbekova@fedora lab10]$ ./lab10-2
Input your name: Aliya
[arayjdarbekova@fedora lab10]$ cat name.txt
My name is:Aliya
[arayjdarbekova@fedora lab10]$ ls -l
итого 52
-rw-----. 1 arayjdarbekova arayjdarbekova 3773 дек 23 10:28 in_out.asm
-rw-r--r--. 1 arayjdarbekova arayjdarbekova 9164 дек 23 10:29 lab10-1
-rwx--x--x. 1 arayjdarbekova arayjdarbekova 1140 дек 23 10:28 lab10-1.asm
-rw-r--r--. 1 arayjdarbekova arayjdarbekova 1472 дек 23 10:29 lab10-1.o
-rwxr-xr-x. 1 arayjdarbekova arayjdarbekova 9144 дек 23 10:34 lab10-2
-rw-----. 1 arayjdarbekova arayjdarbekova 900 дек 23 10:28 lab10-2.asm
-rw-r--r--. 1 arayjdarbekova arayjdarbekova 1552 дек 23 10:34 lab10-2.o
-rwxr-xr-x. 1 arayjdarbekova arayjdarbekova 17 дек 23 10:34 name.txt
-rw-rwxr--. 1 arayjdarbekova arayjdarbekova 0 дек 23 10:28 readme-2.txt
-r-xrwxrwx. 1 arayjdarbekova arayjdarbekova 6 дек 23 10:30 readme.txt

```

Рис. 4.7: Компиляция текста программы lab10-2.asm

5 Выводы

Освоили работу с файлами и правами доступа.