**РОССИЙСКИЙ** **УНИВЕРСИТЕТ** **ДРУЖБЫ**

**НАРОДОВ**

**Факультет** **физико-математических** **и** **естественных**

**наук**

**направление:** **Компьютерные** **и** **информационные**

**науки**

**Лабораторная** **работа** **№10**

**дисциплина:** **Архитектура** **компьютеров** **и** **операционные** **системы**

**студент:** **Гробман** **Александр** **Евгеньевич**

**Группа:** **НКАбд-02-23**

**Цель** **работы**

Целью работы является приобретение навыков написания программ для работы с

файлами.

**Задание**

1. Изучение подпрограмм в ассемблере

2. Освоение возможностей отладчика GDB

3. Рассмотрение примеров работы с отладчиком

4. Выполнение заданий для самостоятельной работы

**Теория**

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. Владельцем 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 rwx,

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 х —

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.

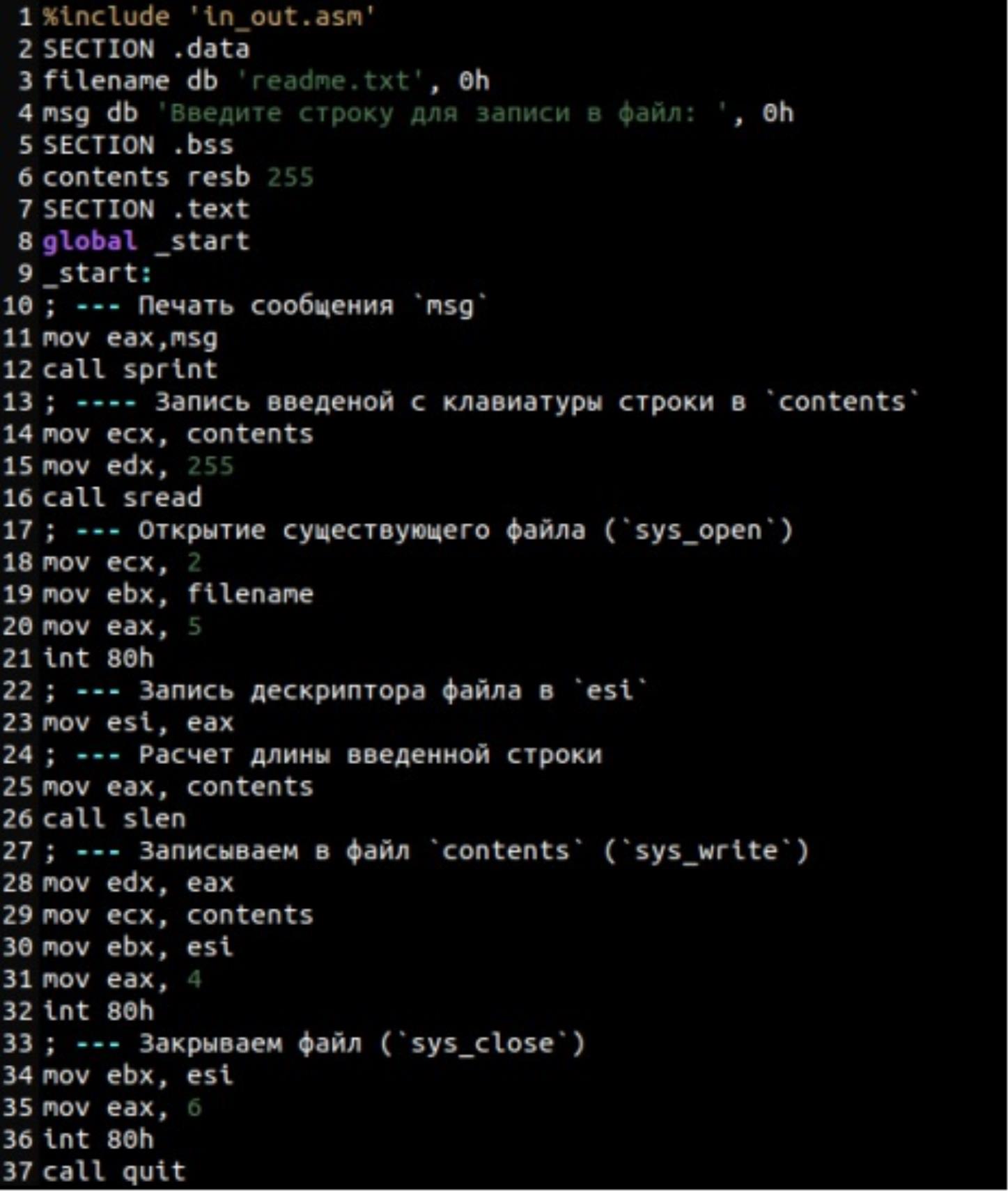
Обработка файлов в операционной системе Linux осуществляется за счет использования

определенных системных вызовов. Для корректной работы и доступа к файлу при его от-

крытии или создании, файлу присваивается уникальный номер (16-битное целое число) –

дескриптор файла

**Выполнение** **лабораторной** **работы**



Чтобы отменить возможность выполнения исполняемого файла lab10-1, я использовал

команду chmod для изменения прав доступа. Я удалил атрибут “x”

во всех трех позициях. После этого я попытался запустить файл, однако он не

запустился, так как отсутствие атрибута “x” во всех трех позициях запрещает

его выполнение.

Я изменил настройки доступа к файлу lab10-1.asm, добавив разрешение на

его выполнение с помощью команды chmod. После этого я попытался запустить файл.

Однако, поскольку это файл с кодом на языке ассемблера, а не команды для терминала,

возникли ошибки. Тем не менее, если в такой файл добавить команды командной

строки, их можно будет выполнить, запустив файл.

Далее, я настроил права доступа к файлам readme в соответствии с указаниями,

представленными в таблице 10.4. Для проверки корректности выполнения,

я использовала команду ls -l.

для варианта 8: rw- -wx --x 010 001 000

**Задание** **для** **самостоятельной** **работы**

Написала программу работающую по следующему алгоритму:

• Вывод приглашения “Как Вас зовут?”

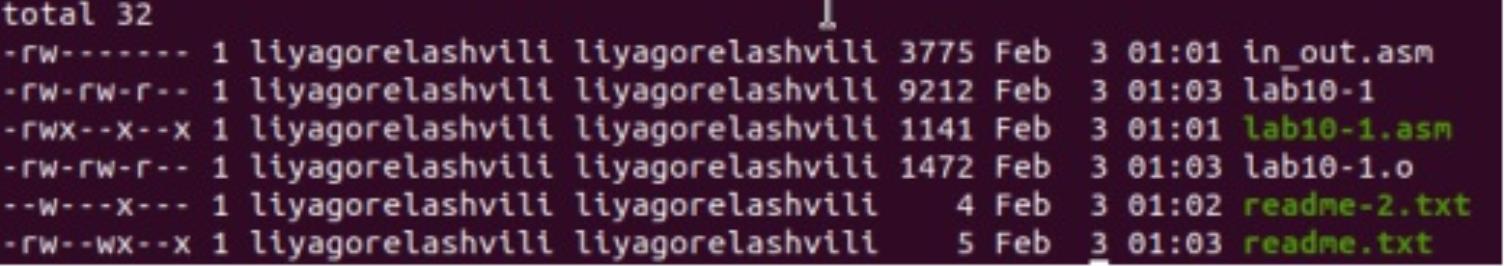
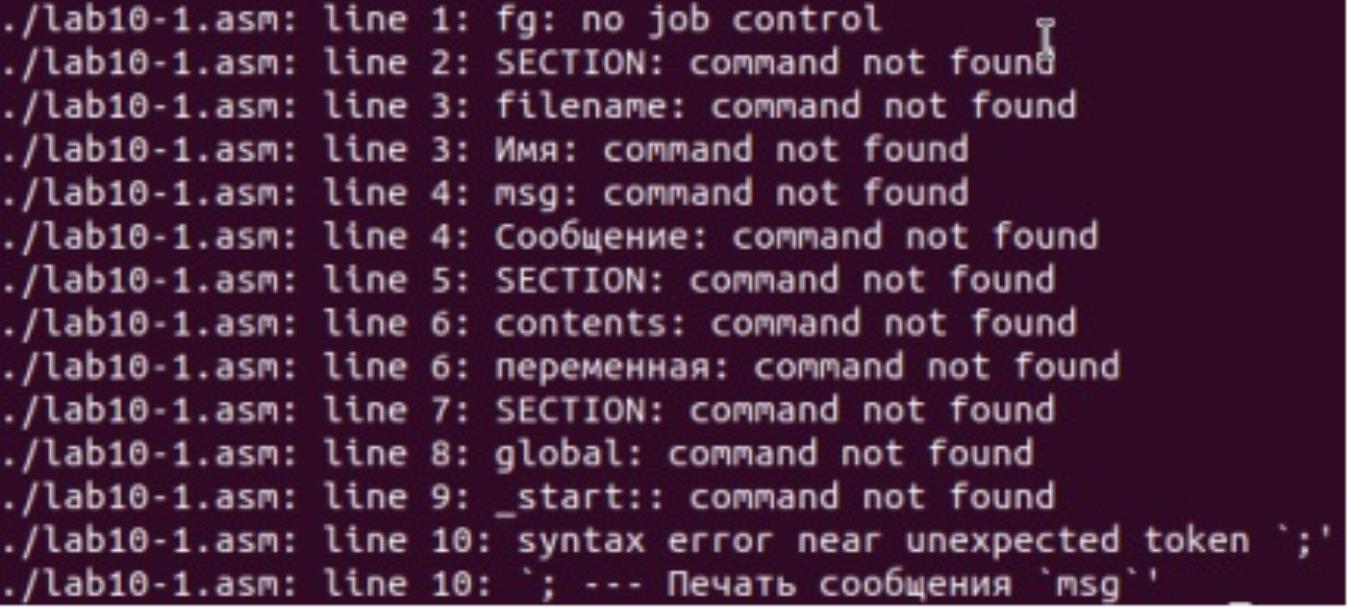
• ввести с клавиатуры свои фамилию и имя

• создать файл с именем name.txt

• записать в файл сообщение “Меня зовут”

• дописать в файл строку введенную с клавиатуры

• закрыть файл



5. **Вывод**

Освоили работy с файлами и правами доступа

**Отправляем** **файлы** **на** **гитхаб.**

Ссылка на отчёт https://github.com/DaOneme/AEGrobman\_study\_2023-

2024\_arhpc/tree/main/Labs/Lab10 (https://github.com/DaOneme/AEGrobman\_study\_2023-

2024\_arhpc/tree/main/Labs/Lab10)

