

Отчет по лабораторной работе №4

Дисциплина: архитектура компьютера

Дельгадильо Валерия

Содержание

1	Цель работы	5
2	Теоретическое введение	6
2.1	Основные принципы работы компьютера	6
2.2	Ассемблер и язык ассемблера	6
3	Лабораторной работы	8
3.1	Программа Hello world!	8
3.2	Транслятор NASM	12
3.3	Расширенный синтаксис командной строки NASM	12
3.4	Компоновщик LD	13
4	Задание для самостоятельной работы	16
5	Выводы	21
6	Список литературы	22

Список иллюстраций

3.1	8
3.2	9
3.3	9
3.4	10
3.5	11
3.6	12
3.7	13
3.8	14
3.9	14
3.10	15
4.1	16
4.2	17
4.3	18
4.4	18
4.5	19
4.6	20

Список таблиц

1 Цель работы

Освоение процедуры компиляции и сборки программ, написанных на ассемблере NASM.

2 Теоретическое введение

2.1 Основные принципы работы компьютера

Основными функциональными элементами любой электронно-вычислительной машины (ЭВМ) являются центральный процессор, память и периферийные устройства.

Взаимодействие этих устройств осуществляется через общую шину, к которой они подключены. Физически шина представляет собой большое количество проводников, соединяющих устройства друг с другом. В современных компьютерах проводники выполнены в виде электропроводящих дорожек на материнской (системной) плате.

2.2 Ассемблер и язык ассемблера

Язык ассемблера (assembly language, сокращённо asm) — машинно-ориентированный язык низкого уровня. Можно считать, что он больше любых других языков приближен к архитектуре ЭВМ и её аппаратным возможностям, что позволяет получить к ним более полный доступ, нежели в языках высокого уровня, таких как C/C++, Perl, Python и пр. Заметим, что получить полный доступ к ресурсам компьютера в современных архитектурах нельзя, самым низким уровнем работы прикладной программы является обращение напрямую к ядру операционной системы. Именно на этом уровне и работают программы,

написанные на ассемблере. Но в отличие от языков высокого уровня ассемблерная программа содержит только тот код, который ввёл программист. Таким образом язык ассемблера — это язык, с помощью которого понятным для человека образом пишутся команды для процессора.

3 Лабораторной работы

3.1 Программа Hello world!

Создайте каталог для работы с программами на языке ассемблера NASM:

```
mkdir -p ~/work/arch-pc/lab04
```

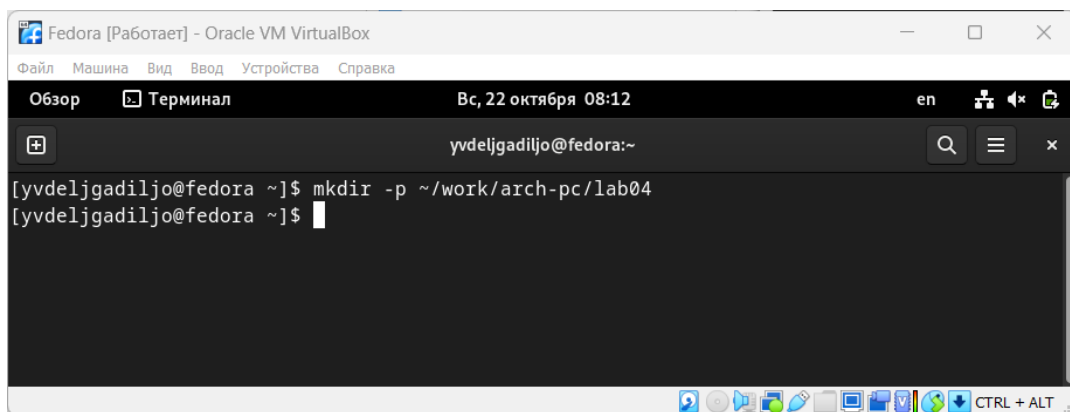


Рис. 3.1:

Перейдите в созданный каталог

```
cd ~/work/arch-pc/lab04
```

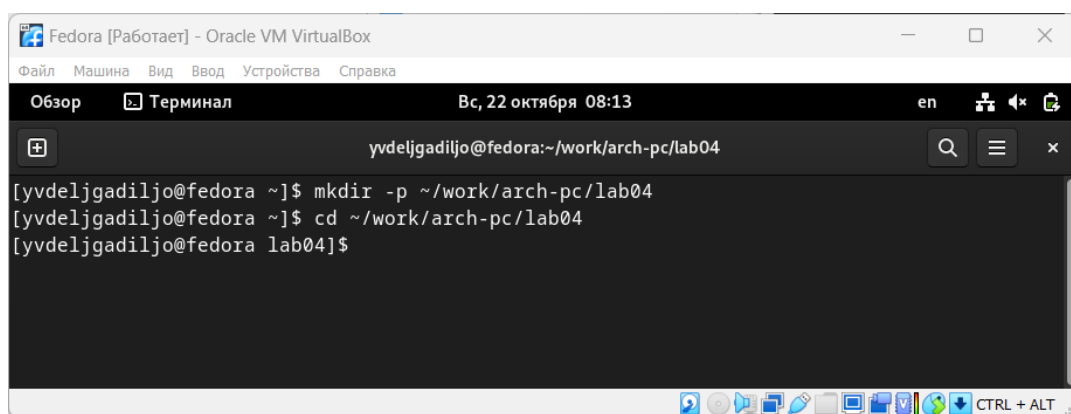



Рис. 3.2:

Создайте текстовый файл с именем hello.asm

touch hello.asm

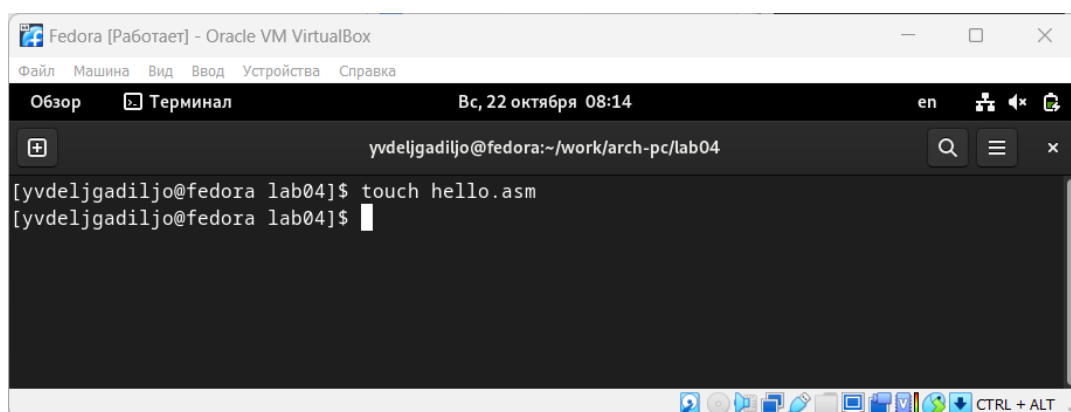


Рис. 3.3:

откройте этот файл с помощью любого текстового редактора, например, gedit

gedit hello.asm

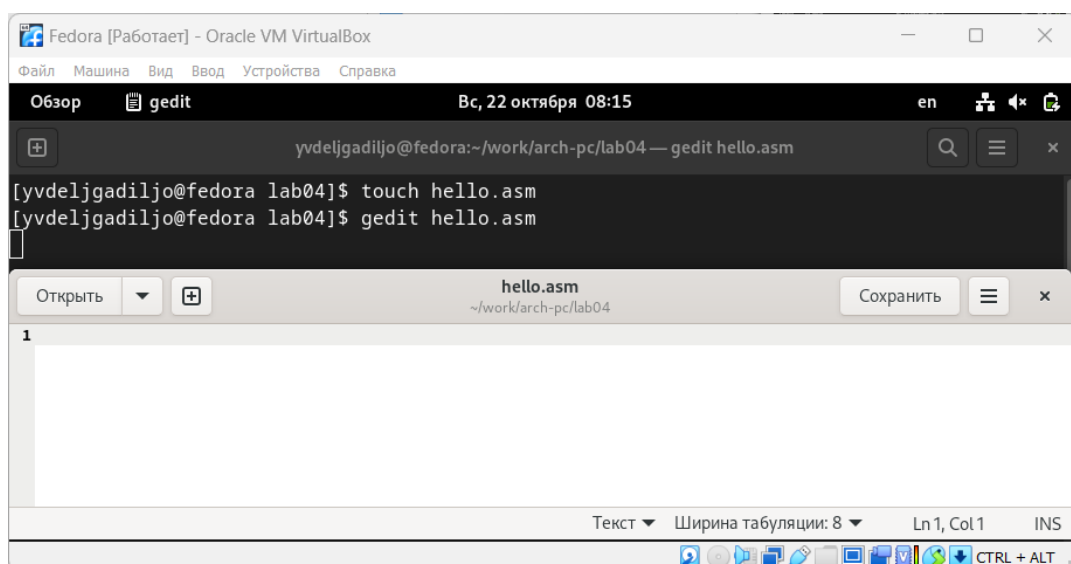


Рис. 3.4:

и введите в него следующий текст:

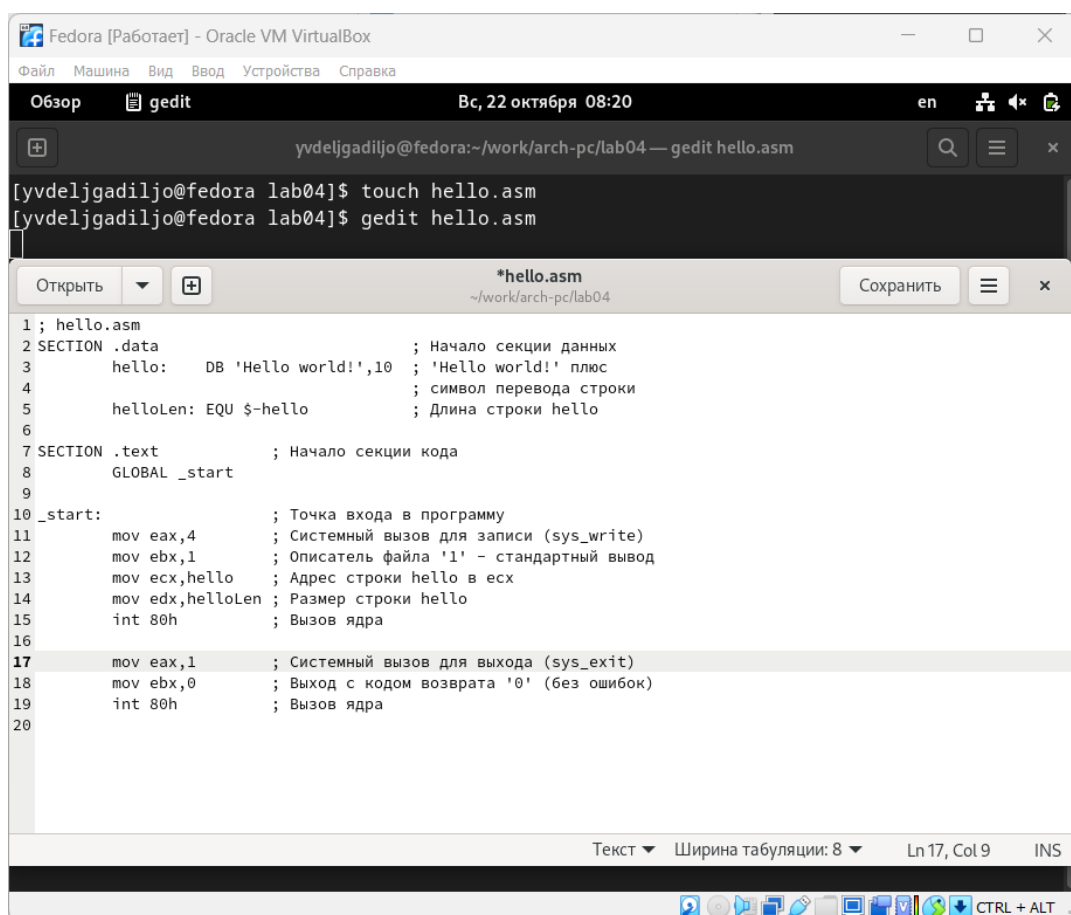


Рис. 3.5:

- В отличие от многих современных высокоуровневых языков программирования, в ассемблерной программе каждая команда располагается на отдельной строке. Размещение нескольких команд на одной строке недопустимо. Синтаксис ассемблера NASM является чувствительным к регистру, т.е. есть разница между большими и малыми буквами.

3.2 Транслятор NASM

NASM превращает текст программы в объектный код. Для компиляции приведённого выше текста программы «Hello World» необходимо написать:

```
nasm -f elf hello.asm
```

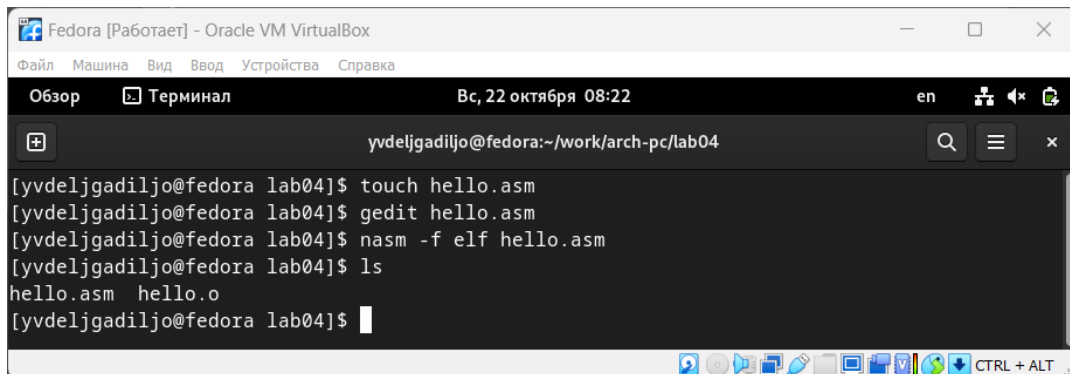


Рис. 3.6:

Текст программы набран без ошибок

3.3 Расширенный синтаксис командной строки NASM

Выполните следующую команду:

```
nasm -o obj.o -f elf -g -l list.lst hello.asm
```

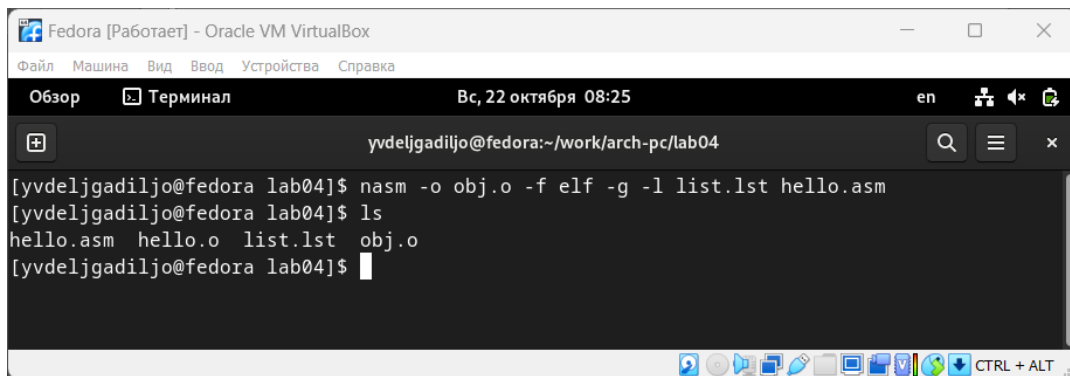


Рис. 3.7:

Данная команда скомпилирует исходный файл `hello.asm` в `obj.o` (опция `-o` позволяет задать имя объектного файла, в данном случае `obj.o`), при этом формат выходного файла будет `elf`, и в него будут включены символы для отладки (опция `-g`), кроме того, будет создан файл листинга `list.lst` (опция `-l`).

С помощью команды `ls` проверяю, что файлы были созданы.

3.4 Компоновщик LD

Объектный файл необходимо передать на обработку компоновщику:

```
ld -m elf_i386 hello.o -o hello
```

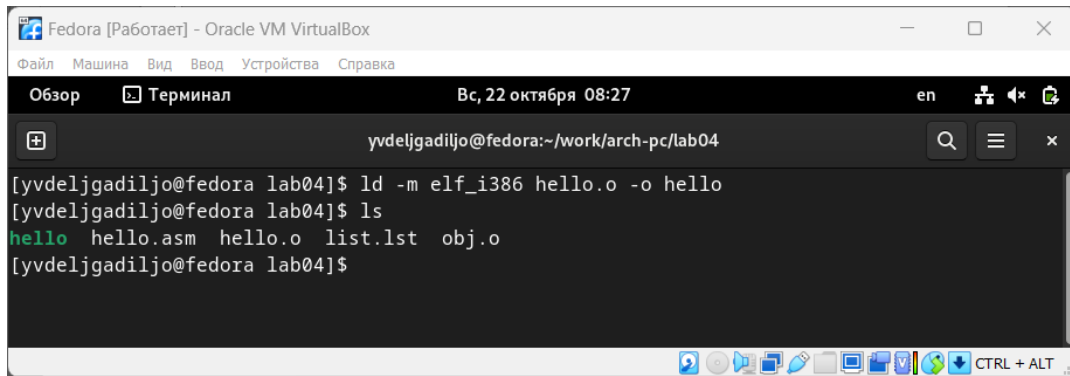


Рис. 3.8:

С помощью команды `ls` проверяю, что исполняемый файл `hello` был создан.

Ключ `-o` с последующим значением задаёт в данном случае имя создаваемого исполняемого файла.

Выполните следующую команду:

```
ld -m elf_i386 obj.o -o main
```

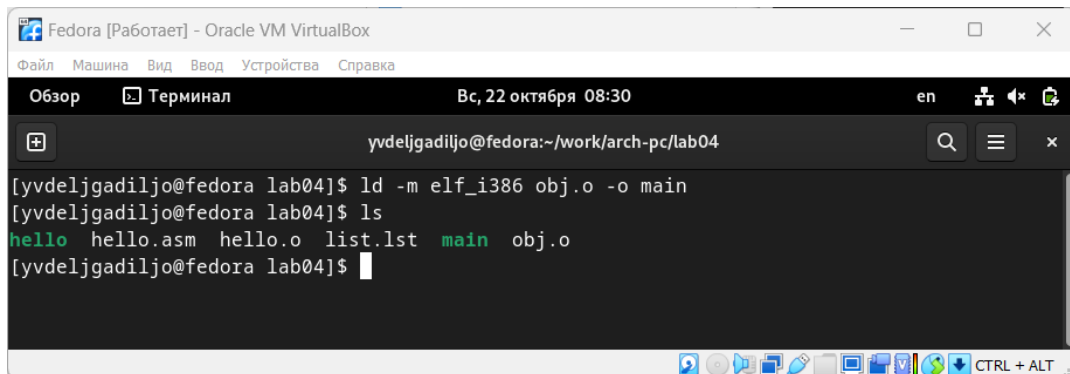


Рис. 3.9:

Запустить на выполнение созданный исполняемый файл, находящийся в текущем каталоге, можно, набрав в командной строке: `./hello`

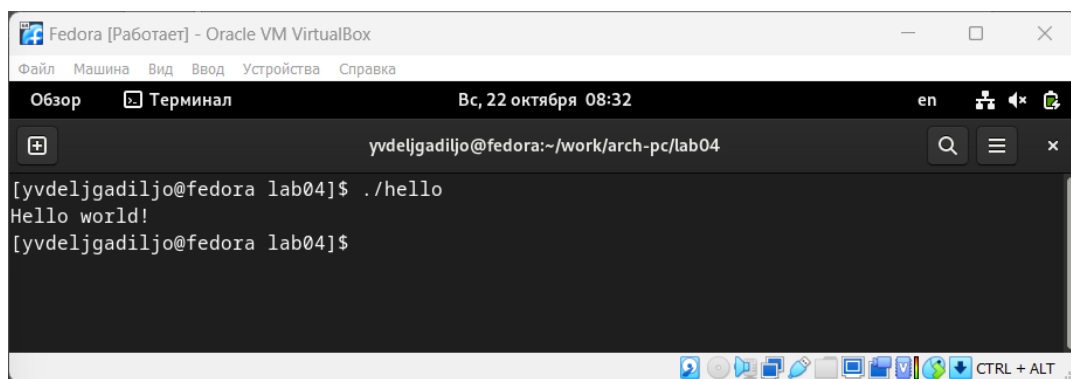
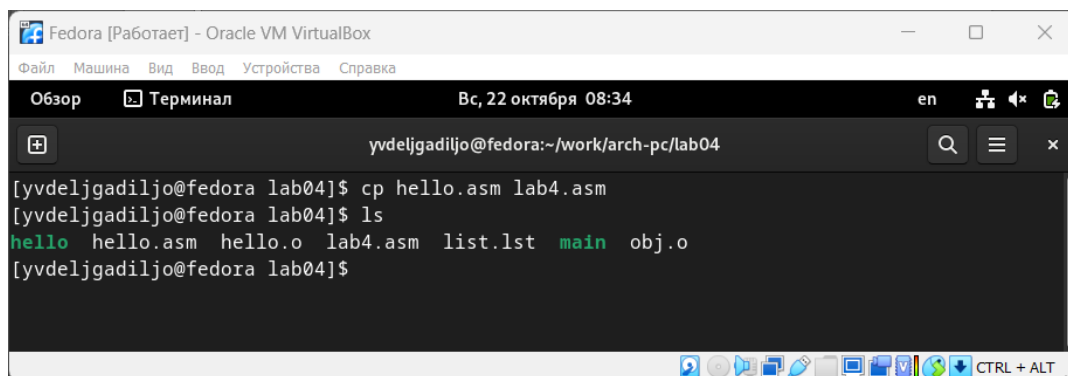


Рис. 3.10:

4 Задание для самостоятельной работы

В каталоге `~/work/arch-pc/lab04` с помощью команды `cp` создайте копию файла

`hello.asm` с именем `lab4.asm`



The screenshot shows a terminal window titled "Fedora [Работает] - Oracle VM VirtualBox". The terminal prompt is `yvdeljgadiljo@fedora:~/work/arch-pc/lab04`. The user has entered the command `cp hello.asm lab4.asm`, followed by `ls`. The output of `ls` is `hello hello.asm hello.o lab4.asm list.lst main obj.o`. The terminal window has a menu bar with "Файл", "Машина", "Вид", "Ввод", "Устройства", and "Справка". The status bar at the bottom shows "CTRL + ALT" and a system tray with various icons.

```
[yvdeljgadiljo@fedora lab04]$ cp hello.asm lab4.asm
[yvdeljgadiljo@fedora lab04]$ ls
hello hello.asm hello.o lab4.asm list.lst main obj.o
[yvdeljgadiljo@fedora lab04]$
```

Рис. 4.1:

С помощью любого текстового редактора внесите изменения в текст программы в файле `lab4.asm` так, чтобы вместо `Hello world!` на экран выводилась строка с вашими фамилией и именем

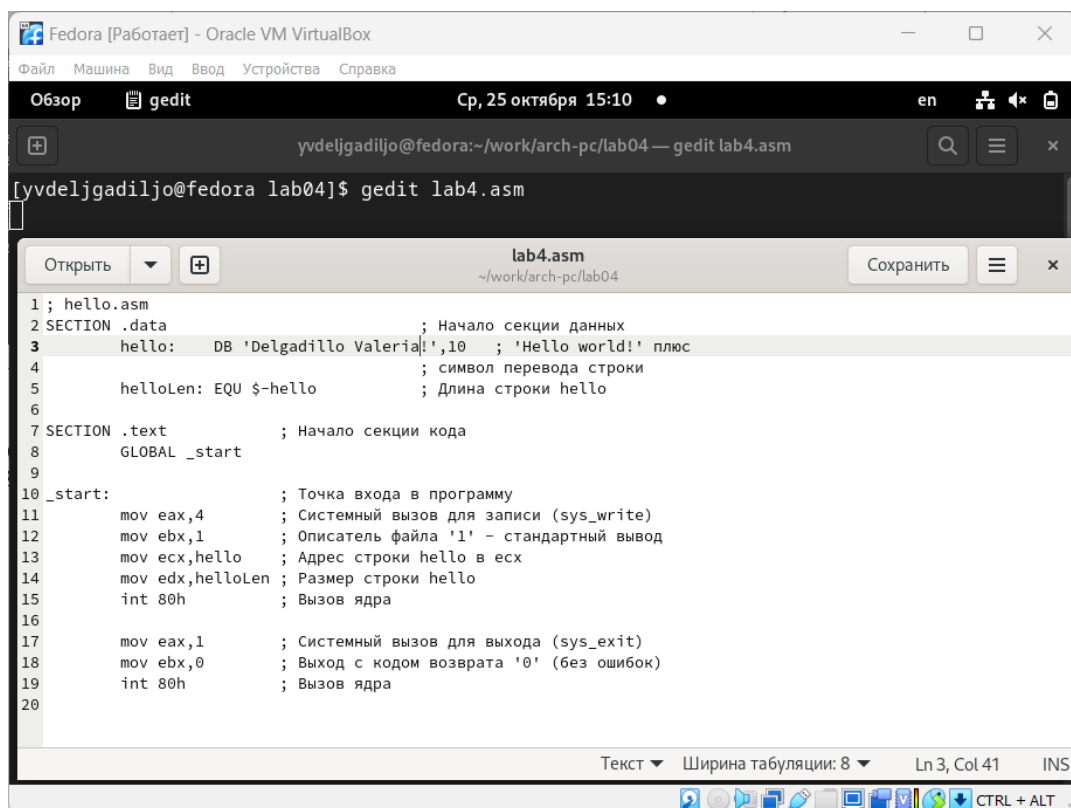
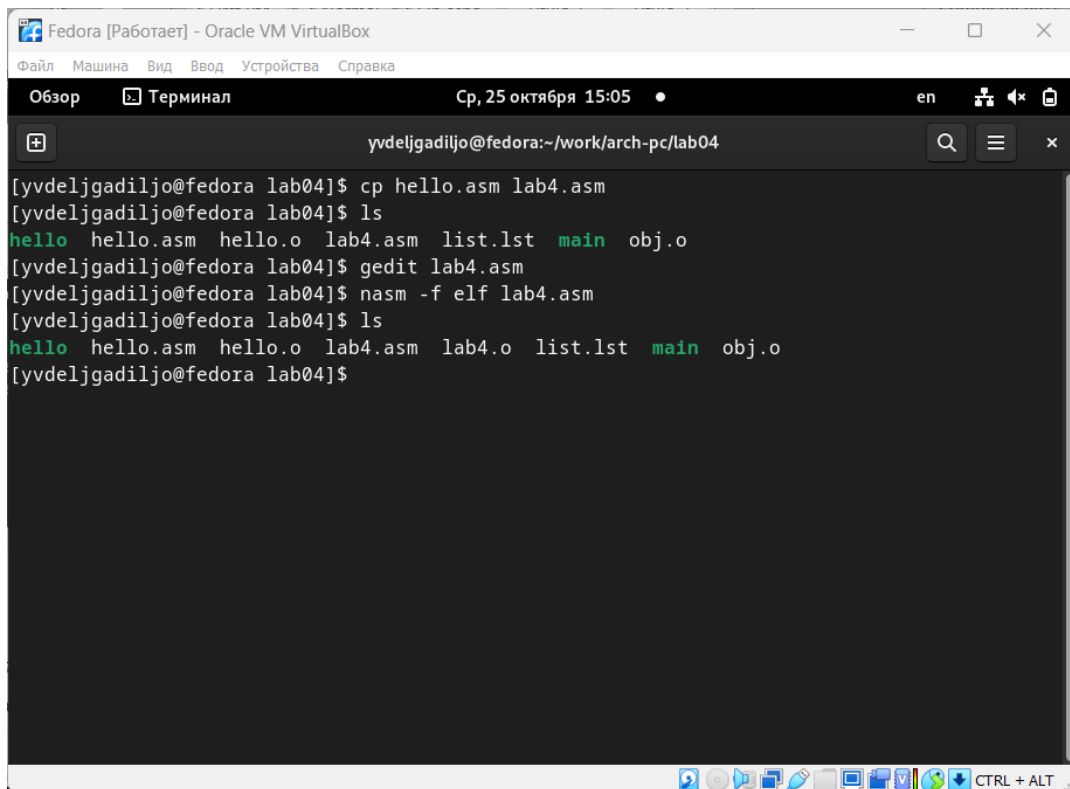


Рис. 4.2:

Оттранслируйте полученный текст программы lab4.asm в объектный файл.

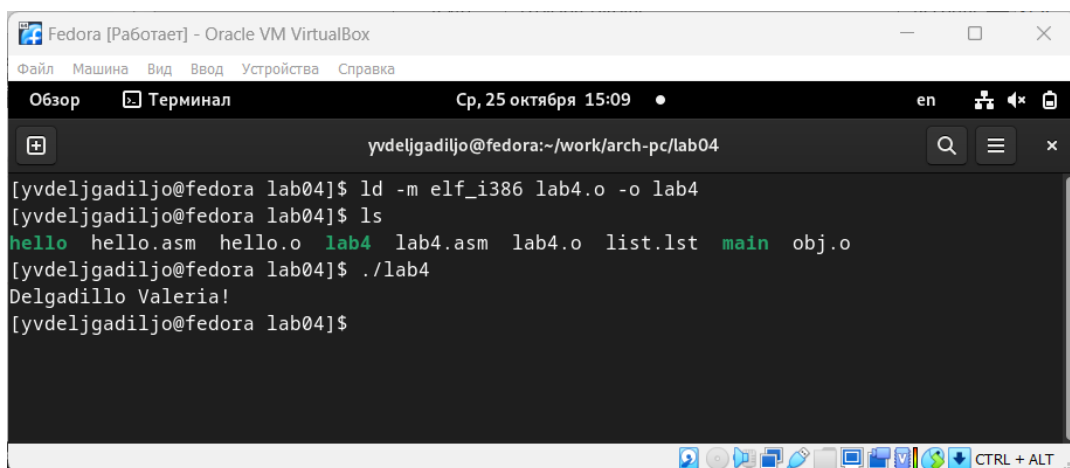


```
Fedora [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
Обзор  Терминал  Ср, 25 октября 15:05  en  [иконки]
yvdeljgadiljo@fedora:~/work/arch-pc/lab04

[yvdeljgadiljo@fedora lab04]$ cp hello.asm lab4.asm
[yvdeljgadiljo@fedora lab04]$ ls
hello  hello.asm  hello.o  lab4.asm  list.lst  main  obj.o
[yvdeljgadiljo@fedora lab04]$ gedit lab4.asm
[yvdeljgadiljo@fedora lab04]$ nasm -f elf lab4.asm
[yvdeljgadiljo@fedora lab04]$ ls
hello  hello.asm  hello.o  lab4.o  list.lst  main  obj.o
[yvdeljgadiljo@fedora lab04]$
```

Рис. 4.3:

Выполните компоновку объектного файла и запустите получившийся исполняемый файл.

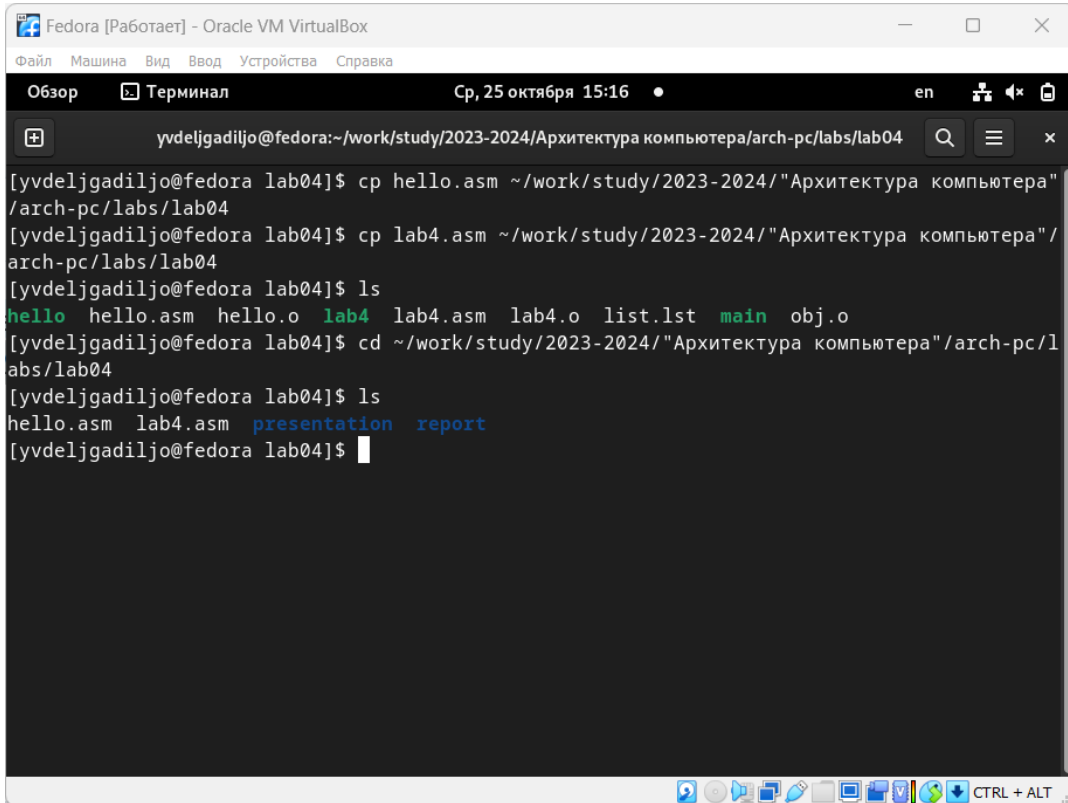


```
Fedora [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
Обзор  Терминал  Ср, 25 октября 15:09  en  [иконки]
yvdeljgadiljo@fedora:~/work/arch-pc/lab04

[yvdeljgadiljo@fedora lab04]$ ld -m elf_i386 lab4.o -o lab4
[yvdeljgadiljo@fedora lab04]$ ls
hello  hello.asm  hello.o  lab4  lab4.asm  lab4.o  list.lst  main  obj.o
[yvdeljgadiljo@fedora lab04]$ ./lab4
Delgadillo Valeria!
[yvdeljgadiljo@fedora lab04]$
```

Рис. 4.4:

Скопируйте файлы `hello.asm` и `lab4.asm` в Ваш локальный репозиторий в каталог `~/work/study/2023-2024/”Архитектура компьютера”/arch-pc/labs/lab04/`.

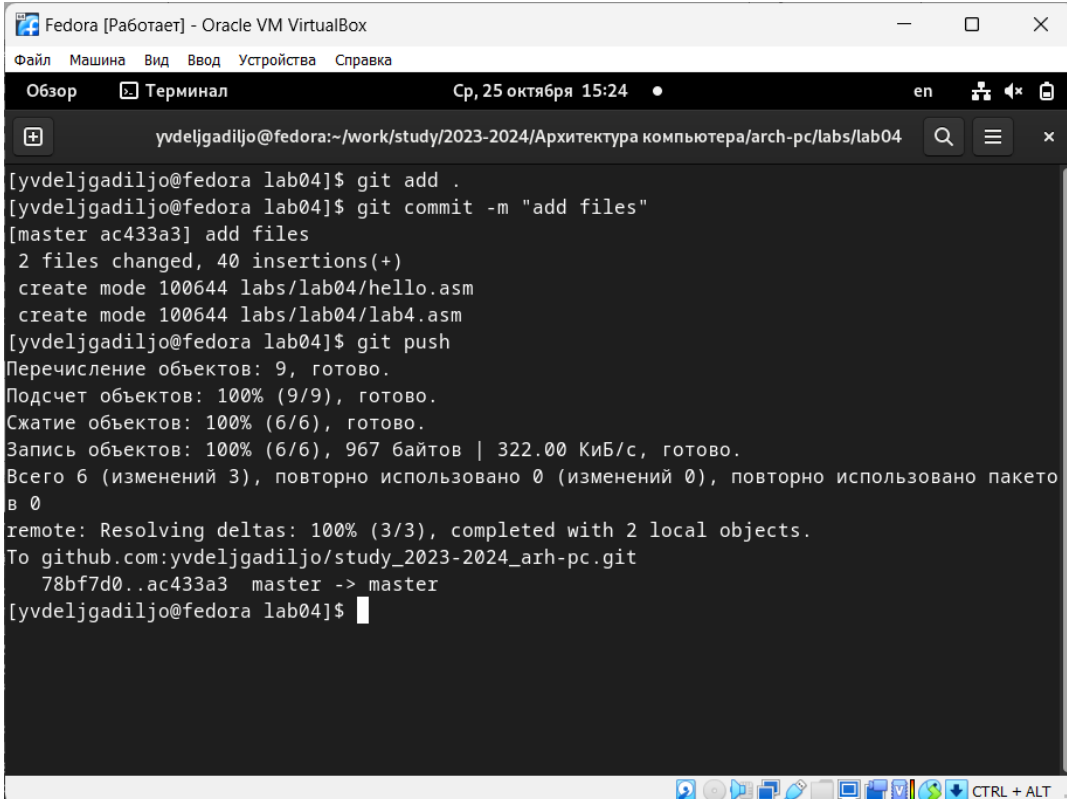


The screenshot shows a terminal window titled "Fedora [Работает] - Oracle VM VirtualBox". The terminal displays the following commands and output:

```
yvdeljgatiljo@fedora:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab04
[yvdeljgatiljo@fedora lab04]$ cp hello.asm ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab04
[yvdeljgatiljo@fedora lab04]$ cp lab4.asm ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab04
[yvdeljgatiljo@fedora lab04]$ ls
hello hello.asm hello.o lab4 lab4.asm lab4.o list.lst main obj.o
[yvdeljgatiljo@fedora lab04]$ cd ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab04
[yvdeljgatiljo@fedora lab04]$ ls
hello.asm lab4.asm presentation report
[yvdeljgatiljo@fedora lab04]$
```

Рис. 4.5:

Загрузите файлы на Github.



The image shows a terminal window titled "Fedora [Работает] - Oracle VM VirtualBox". The window has a menu bar with "Файл", "Машина", "Вид", "Ввод", "Устройства", and "Справка". Below the menu bar is a toolbar with "Обзор" and "Терминал" buttons. The terminal shows the following commands and output:

```
[yvdeljgadiljo@fedora lab04]$ git add .
[yvdeljgadiljo@fedora lab04]$ git commit -m "add files"
[master ac433a3] add files
2 files changed, 40 insertions(+)
 create mode 100644 labs/lab04/hello.asm
 create mode 100644 labs/lab04/lab4.asm
[yvdeljgadiljo@fedora lab04]$ git push
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 967 байтов | 322.00 КиБ/с, готово.
Всего 6 (изменений 3), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (3/3), completed with 2 local objects.
To github.com:yvdeljgadiljo/study_2023-2024_arh-pc.git
 78bf7d0..ac433a3 master -> master
[yvdeljgadiljo@fedora lab04]$
```

The terminal window also shows a search bar at the top right with a magnifying glass icon and a close button. The bottom status bar shows various system icons and the text "CTRL + ALT".

Рис. 4.6:

5 Выводы

В ходе данной лабораторной работы я освоила процедуры компиляции и сборки программ, написанных на ассемблере NASM.

6 Список литературы

- GDB: The GNU Project Debugger. — URL: <https://www.gnu.org/software/gdb/>.
- GNU Bash Manual. — 2016. — URL: <https://www.gnu.org/software/bash/manual/>.
- Midnight Commander Development Center. — 2021. — URL: <https://midnight-commander.org/>.
- NASM Assembly Language Tutorials. — 2021. — URL: <https://asmtutor.com/>.
- 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/Learningbash-Shell-Programming-Nutshell/dp/0596009658>.
- Robbins A. Bash Pocket Reference. — O'Reilly Media, 2016. — 156 с. — ISBN 978-1491941591.
- The NASM documentation. — 2021. — URL: <https://www.nasm.us/docs.php>.
- Zarrelli G. Mastering Bash. — Packt Publishing, 2017. — 502 с. — ISBN 9781784396879.
- Колдаев В. Д., Лупин С. А. Архитектура ЭВМ. — М. : Форум, 2018.
- Куляс О. Л., Никитин К. А. Курс программирования на ASSEMBLER. — М. : Солон-Пресс, 2017.
- Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
- Расширенный ассемблер: NASM. — 2021. — URL: <https://www.opennet.ru/docs/RUS/nasm/>.

- Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВПетербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.
- 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.
- Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. — 874 с. — (Классика Computer Science).
- Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер, 2015. — 1120 с. — (Классика Computer Science).