ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 4

дисциплина: Архитектура компьютера

Козин Иван Евгеньевич

Нкабд-03-25

Содержание

Цель работы

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

---

2. Порядок выполнения лабораторной работы

В "Таблице 2.1"приведены основные команды, используемые при работе с Git и компиляцией отчетов.

Таблица 2.1: Основные команды, используемые при выполнении работы

Команда	Флаги	Назначение
nasm	-f elf	Компиляция в объектный файл формата ELF
	-o file.o	Указание имени выходного объектного файла
	-g	Включение отладочной информации
	-l list.lst	Создание файла листинга
ld	-m elf_i386	Сборка для 32-битной архитектуры
	-o executable_file	Указание имени исполняемого файла

2.1 Программа Hello world!

Создадим каталог для работы с программами на языке ассемблера NASM

```
iekozin@fedora:~$ mkdir -p ~/work/study/2025-2026/Архитектура\ компьютера/arch-pc/labs/lab04
```

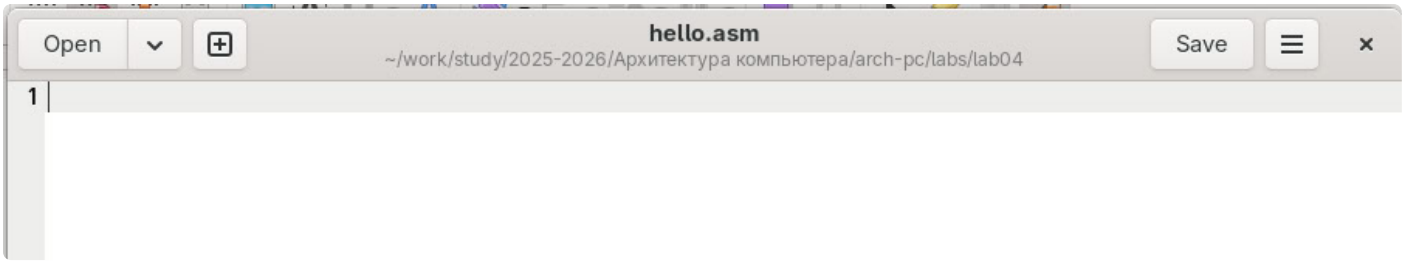
Перейдем в созданный каталог

```
iekozin@fedora:~$ cd ~/work/study/2025-2026/Архитектура\ компьютера/arch-pc/labs/lab04
```

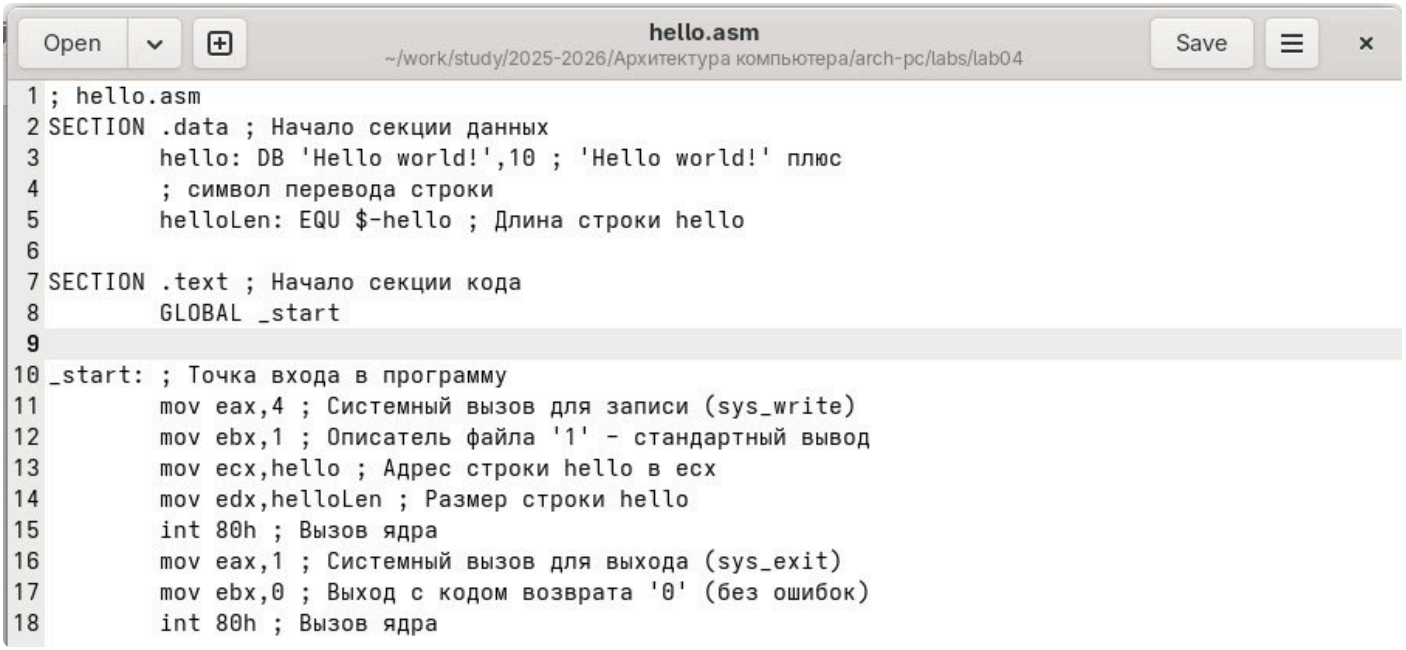
Создадим текстовый файл с именем hello.asm и проверим его наличие

```
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ touch hello.asm
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello.asm  presentation  report
```

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



Введем в него следующий текст



2.1 Транслятор NASM

Для компиляции приведённого выше текста программы «Hello World» необходимо написать nasm -f elf hello.asm и проверим объектный файл был создан

```
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ nasm -f elf hello.asm
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello.asm  hello.o  presentation  report
```

2.3 Расширенный синтаксис командной строки NASM

Выполним следующую команду nasm -o obj.o -f elf -g -l list.lst hello.asm и проверим его наличие в каталоге

```
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$  nasm -o obj.o -f elf -g -l list.
lst hello.asm
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello.asm  hello.o  list.lst  obj.o  presentation  report
```

2.4 Компоновщик LD

Объектный файл передадим на обработку компоновщику и проверим

```
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ ld -m elf_i386 hello.o -o hello
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello  hello.asm  hello.o  list.lst  obj.o  presentation  report
```

Выполним следующую команду ld -m elf_i386 obj.o -o main и проверим

```
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ ld -m elf_i386 obj.o -o main
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello  hello.asm  hello.o  list.lst  main  obj.o  presentation  report
```

2.5 Запуск исполняемого файла

Наберем в командной строке ./hello и проверим что выводит

```
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ ./hello
Hello world!
```

3 Задание для самостоятельной работы

3.1 1 Задание

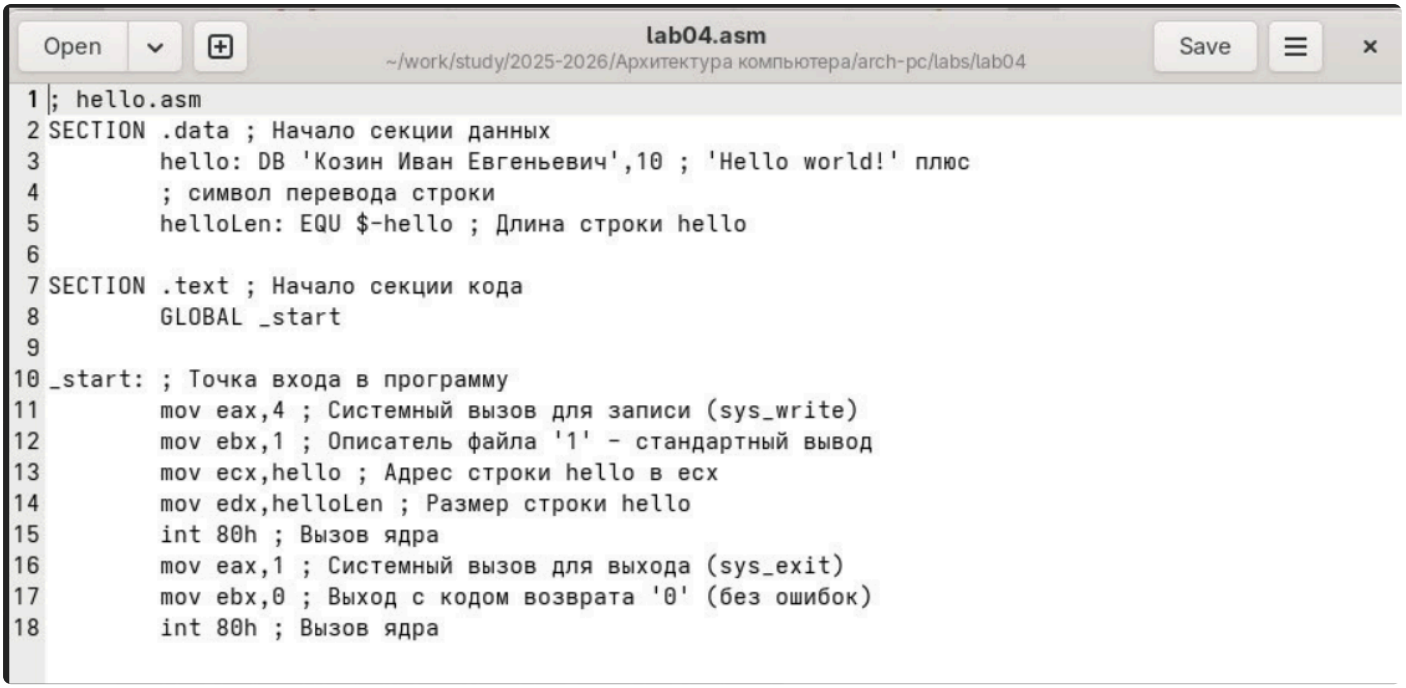
Создайте копию файла hello.asm с именем lab4.asm.

```
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ cp hello.asm lab04.asm
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello  hello.asm  hello.o  lab04.asm  list.lst  main  obj.o  presentation  report
```

3.2 2 Задание

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.

С помощью gedit открываем и редактируем файл



3.3 3 Задание

Оттранслируйте полученный текст программы lab4.asm в объектный файл. Вы- полните компоновку объектного файла и запустите получившийся исполняемый файл.

Оттранслируем полученный текст программы lab4.asm в объектный файл и про- верим файл

```
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ nasm -f elf lab04.asm
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello  hello.asm  hello.o  lab04.asm  lab04.o  list.lst  main  obj.o  presentation  report
```

Выполним компоновку объектного файла

```
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ ld -m elf_i386 lab04.o -o lab04
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello  hello.asm  hello.o  lab04  lab04.asm  lab04.o  list.lst  main  obj.o  presentation  report
```

Запустим получившийся исполняемый файл

```
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ ./lab04
Козин Иван Евгеньевич
```

3.4 4 Задание

На нашем локальном репозитории уже лежат следующий файлы(которые были созданы по мере проведения лабораторной работы):

1. Файлы hello и все к ним прилежащие
2. Файлы lab04 и все к ним прилежащие

```
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04/report$ git add .
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04/report$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   ЛБ_04_Козин_Отчёт.docx
        new file:   ЛБ_04_Козин_Отчёт.md
        new file:   ЛБ_04_Козин_Отчёт.pdf

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        ../hello
        ../hello.asm
        ../hello.o
        ../lab04
        ../lab04.asm
        ../lab04.o
        ../list.lst
        ../main
        ../obj.o

iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04/report$ git commit -m "Commit for 04 lab"
[master bc5c837] Commit for 04 lab
 3 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 labs/lab04/report/ЛБ_04_Козин_Отчёт.docx
 create mode 100644 labs/lab04/report/ЛБ_04_Козин_Отчёт.md
 create mode 100644 labs/lab04/report/ЛБ_04_Козин_Отчёт.pdf
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04/report$ git push -u origin
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 513 bytes | 513.00 KiB/s, done.
```

```
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04/report$ cd ..
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ git add .
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   hello
        new file:   hello.asm
        new file:   hello.o
        new file:   lab04
        new file:   lab04.asm
        new file:   lab04.o
        new file:   list.lst
        new file:   main
        new file:   obj.o

iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ git commit -m "files for 04laba .asm and another"
[master 7686e83] files for 04laba .asm and another
 9 files changed, 55 insertions(+)
 create mode 100755 labs/lab04/hello
 create mode 100644 labs/lab04/hello.asm
 create mode 100644 labs/lab04/hello.o
 create mode 100755 labs/lab04/lab04
 create mode 100644 labs/lab04/lab04.asm
 create mode 100644 labs/lab04/lab04.o
 create mode 100644 labs/lab04/list.lst
 create mode 100755 labs/lab04/main
 create mode 100644 labs/lab04/obj.o
iekozin@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab04$ git push -u origin
Enumerating objects: 16, done.
Counting objects: 100% (16/16), done.
Delta compression using up to 4 threads
Compressing objects: 100% (13/13), done.
```

3.5 Выводы

В ходе лабораторной работы были освоены основные этапы создания программ на языке ассемблера NASM: написание исходного кода, трансляция с помощью NASM, компоновка объектного файла с использованием ld и запуск исполняемого файла. Были изучены основные директивы ассемблера и системные вызовы Linux для организации ввода-вывода. Полученные навыки позволяют создавать низкоуровневые программы с прямым доступом к системным ресурсам.