

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 4

дисциплина: Архитектура компьютера

Студент: Сухова Арина Евгеньевна

Группа: НПИбд-02-25

МОСКВА

2025 г.

1. Цель работы:

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

2. Задание:

Написать 2 программы (Hello world, lab4(Имя Фамилия))

3. Порядок выполнения работы:

3.1 Программа Hello world!

Создаем каталог для работы с программами на языке ассемблера NASM (рис. 1).

```
aesukhova@fedora:~$ mkdir -p ~/work/arch-pc/lab04
aesukhova@fedora:~$
```

Рис. 1: Создаем каталоги с помощью команды *mkdir*

Переходим в созданный каталог (рис. 2).

```
aesukhova@fedora:~$ cd ~/work/arch-pc/lab04
aesukhova@fedora:~/work/arch-pc/lab04$
```

Рис. 2: Переходим в каталог с помощью команды *cd*

Создаем текстовый файл (рис. 3).

```
aesukhova@fedora:~/work/arch-pc/lab04$ touch hello.asm
aesukhova@fedora:~/work/arch-pc/lab04$
```

Рис. 3: Создаем текстовый файл *hello.asm*

Открываем данный файл в текстовом редакторе (рис. 4).

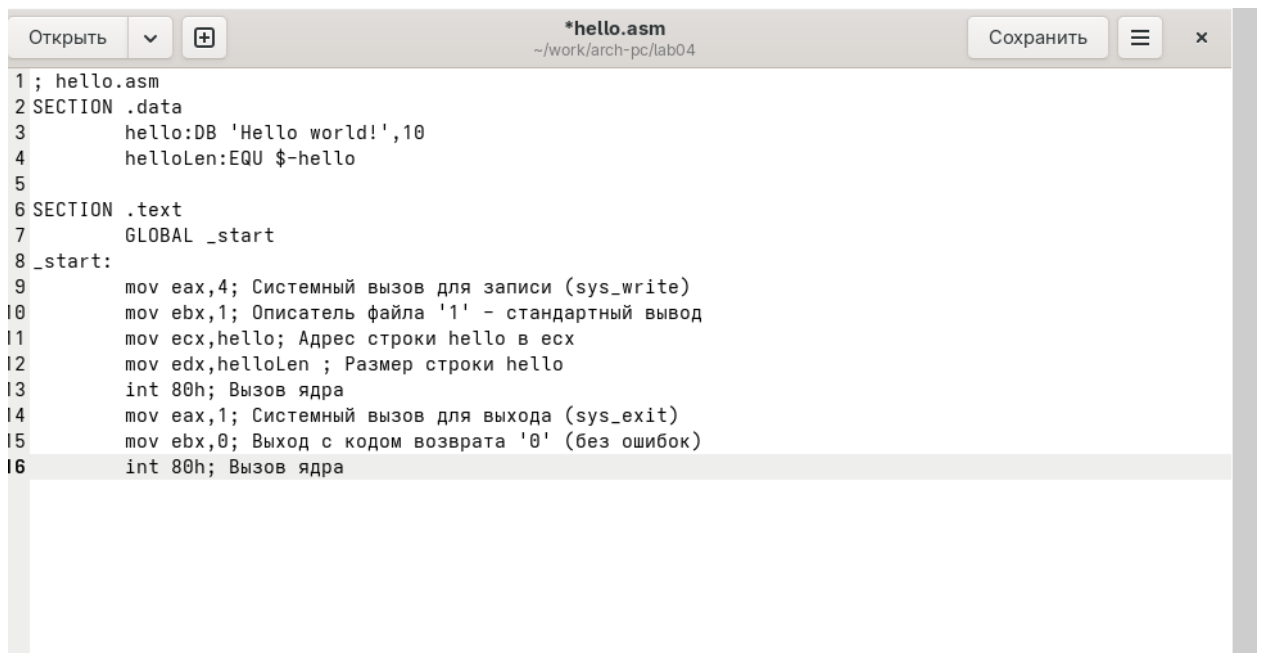


Рис. 4: Открываем файл и заполняем его по примеру

3.2 Транслятор NASM

Преобразуем текст программы в объектный код (рис. 5).

```
aesukhova@fedora:~/work/arch-pc/lab04$ nasm -f elf hello.asm
```

Рис. 5: Используем команду *nasm*

Проверяем создался ли объектный файл с помощью команды *ls* (рис. 6).

```
aesukhova@fedora:~/work/arch-pc/lab04$ ls
hello.asm hello.o
```

Рис. 6: Проверяем работу команды

3.3 Расширенный синтаксис командной строки NASM

Компилируем исходный файл (рис. 7).

```
aesukhova@fedora:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.lst hello.asm
```

Рис. 7: Преобразуем файл *hello.asm* в *obj.o*

Проверяем, как сработала команда (рис. 8).

```
aesukhova@fedora:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.lst hello.asm
aesukhova@fedora:~/work/arch-pc/lab04$ ls
hello.asm hello.o list.lst obj.o
aesukhova@fedora:~/work/arch-pc/lab04$
```

Рис. 8: Проверяем создание файла командой *ls*

3.4 Компоновщик LD

Передаем объектный файл на обработку компоновщику (рис. 9).

```
aesukhova@fedora:~/work/arch-pc/lab04$ ld -m elf_i386 hello.o -o hello
```

Рис. 9: Используем команду *ld*

Проверяем создался ли исполняемый файл *hello* (рис. 10).

```
aesukhova@fedora:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst obj.o
```

Рис. 10: Используем команду *ls*

Передаем объектный файл на обработку компоновщику (рис. 11).

```
aesukhova@fedora:~/work/arch-pc/lab04$ ld -m elf_i386 obj.o -o main
```

Рис. 11: Используем команду *ld*, создавая файл *main*

Проверяем создался ли исполняемый файл *hello* (рис. 12).

```
aesukhova@fedora:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst main obj.o
```

Рис. 12: Используем команду *ls*

3.5 Запуск исполняемого файла

Запускаем на выполнение созданный исполняемый файл (рис. 13).

```
aesukhova@fedora:~/work/arch-pc/lab04$ ./hello
Hello world!
```

Рис. 13: Используем команду `./hello`

3.6 Задание для самостоятельной работы

Создаем копию файла `hello.asm` (рис. 14).

```
aesukhova@fedora:~/work/arch-pc/lab04$ cp hello.asm lab4.asm
```

Рис. 14: Используем команду `cp`

Открываем файл и редактируем его (рис. 15).

```
aesukhova@fedora:~/work/arch-pc/lab04$ gedit lab4.asm
```

Рис. 15: Открываем файл в текстовом редакторе

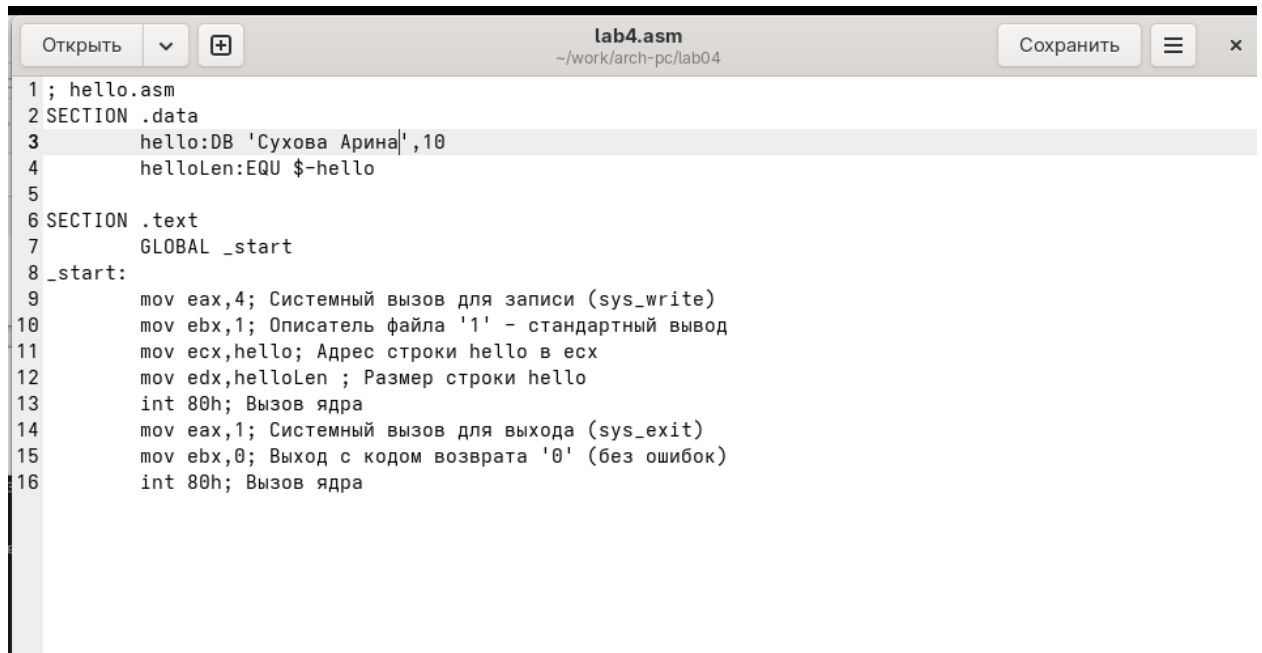


Рис. 16: Редактируем файл для своего имени и фамилии

Прописываем те же команды, что и с первой программой (рис. 17).

```
aesukhova@fedora:~/work/arch-pc/lab04$ nasm -f elf lab4.asm
aesukhova@fedora:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.lst lab4.asm
aesukhova@fedora:~/work/arch-pc/lab04$ ld -m elf_i386 lab4.o -o hello
aesukhova@fedora:~/work/arch-pc/lab04$ ld -m elf_i386 obj.o -o main
aesukhova@fedora:~/work/arch-pc/lab04$ ./hello
Сухова Арина
```

Рис. 17: Прописываем команды для работы файла и запускаем программу

Копируем файлы в локальный репозиторий (рис. 18).

```

aesukhova@fedora:~/work/arch-pc/lab04$ cp hello.asm ~/work/study/2025-2026/"Архитектура компьютера"/arch-
pc/labs/lab04/
aesukhova@fedora:~/work/arch-pc/lab04$ ^C
aesukhova@fedora:~/work/arch-pc/lab04$ cp lab4.asm ~/work/study/2025-2026/"Архитектура компьютера"/arch-p
c/labs/lab04/

```

Рис. 18: Копируем файлы в каталог с ЛР4

Переходим в каталог лабораторных работ и загружаем файлы на Github (рис. 19).

```

[spelovandrei@fedora lab04]$ cd ~/work/study/2023-2024/"Архитектура компьютера"/
arch-pc
[spelovandrei@fedora arch-pc]$ git add .
[spelovandrei@fedora arch-pc]$ git commit -am 'feat(main): add files lab-4'
[master d70625c] feat(main): add files lab-4
 2 files changed, 34 insertions(+)
 create mode 100644 labs/lab04/hello.asm
 create mode 100644 labs/lab04/lab4.asm
[spelovandrei@fedora arch-pc]$ git push
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 8 потоков
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 979 байтов | 979.00 КиБ/с, готово.
Всего 6 (изменений 3), повторно использовано 0 (изменений 0), повторно использов
ано пакетов 0
remote: Resolving deltas: 100% (3/3), completed with 2 local objects.
To github.com:SpelovA/study_2023-2024_arh-pc.git
   82f0b12..d70625c  master -> master
[spelovandrei@fedora arch-pc]$

```

Рис. 19: Загружаем файлы

4 Выводы

Мы познакомились с языком ассемблера NASM и создали две работающих программы.