

# **Шаблон отчёта по лабораторной работе**

**Простейший вариант**

Дмитрий Сергеевич Кулябов

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
3.1	Программа Hello world! . . . . .	7
3.2	Транслятор NASM . . . . .	8
3.3	Расширенный синтаксис командной строки NASM . . . . .	8
3.4	Компановщик LD . . . . .	9
3.5	Запуск исполняемого файла . . . . .	9
3.6	Задание для самостоятельной работы . . . . .	10
<b>4</b>	<b>Выводы</b>	<b>12</b>

# Список иллюстраций

3.1	Создаём каталог с помощью команды <code>mkdir</code> . . . . .	7
3.2	Переходим в каталог с помощью команды <code>cd</code> . . . . .	7
3.3	Создаём текстовый файл <code>hello.asm</code> . . . . .	7
3.4	Открываем файл и заполняем его по примеру . . . . .	8
3.5	Используем команду <code>nasm</code> . . . . .	8
3.6	Проверяем работу команды . . . . .	8
3.7	Преобразуем файл <code>hello.asm</code> в <code>obj.o</code> . . . . .	8
3.8	Проверяем создание файла командой <code>ls</code> . . . . .	9
3.9	Используем команду <code>ld</code> . . . . .	9
3.10	Используем команду <code>ls</code> . . . . .	9
3.11	Используем команду <code>ld</code> , создавая файл <code>main</code> . . . . .	9
3.12	Используем команду <code>ls</code> . . . . .	9
3.13	Используем команду <code>./hello</code> . . . . .	10
3.14	Используем команду <code>cp</code> . . . . .	10
3.15	Открываем файл в текстовом редакторе . . . . .	10
3.16	Редактируем файл для вывода своего имени и фамилии . . . . .	10
3.17	Прописываем команды для работы файла и запускаем программу	10
3.18	Копируем файлы в каталог с ЛР4 . . . . .	11
3.19	Загружаем файлы . . . . .	11

## **Список таблиц**

# 1 Цель работы

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

## 2 Задание

1. Создание программы Hello world!
2. Работа с транслятором NASM
3. Работа с расширенным синтаксисом командной строки NASM
4. Работа с компоновщиком LD
5. Запуск исполняемого файла
6. Выполнение заданий для самостоятельной работы.

## 3 Выполнение лабораторной работы

### 3.1 Программа Hello world!

Создаём каталог для работы с программами на языке ассемблера NASM (рис. 3.1).

```
mavinogradova@10:~$ mkdir -p ~/work/arch-pc/lab04  
mavinogradova@10:~$
```

Рис. 3.1: Создаём каталог с помощью команды mkdir

Переходим в созданный каталог (рис. 3.2).

```
mavinogradova@10:~$ cd work/arch-pc/lab04  
mavinogradova@10:~/work/arch-pc/lab04$
```

Рис. 3.2: Переходим в каталог с помощью команды cd

Создаём текстовый файл (рис. 3.3).

```
mavinogradova@10:~/work/arch-pc/lab04$ touch hello.asm  
mavinogradova@10:~/work/arch-pc/lab04$
```

Рис. 3.3: Создаём текстовый файл hello.asm

Открываем данный файл в текстовом редакторе (рис. 3.4).

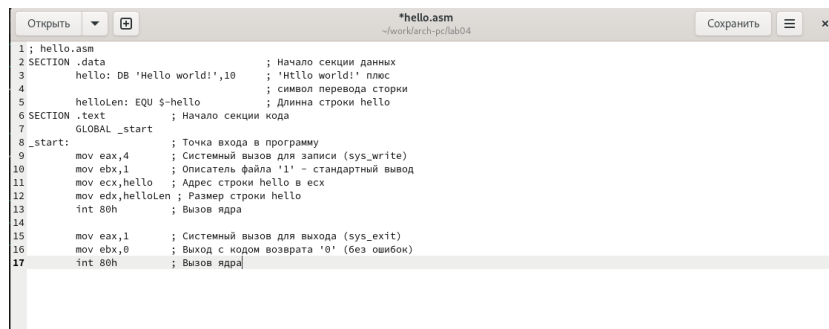


Рис. 3.4: Открываем файл и заполняем его по примеру

## 3.2 Транслятор NASM

Преобразуем текст программы в объектный код (рис. 3.5).

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

Рис. 3.5: Используем команду nasm

Проверяем созданся ли объектный файл с помощью команды ls (рис. 3.6).

```

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

```

Рис. 3.6: Проверяем работу команды

## 3.3 Расширенный синтаксис командной строки NASM

Компилируем исходный файл (рис. 3.7).

```

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

```

Рис. 3.7: Преобразуем файл hello.asm в obj.o

Проверяем как сработала команда (рис. 3.8).



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

Рис. 3.8: Проверяем создание файла командой ls

## 3.4 Компановщик LD

Передаём объектный файл на обработку компановщику (рис. 3.9).

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

Рис. 3.9: Используем команду ld

Проверяем созданся ли исполняемый файл hello (рис. 3.10).

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

Рис. 3.10: Используем команду ls

Передаём объектный файл на обработку компановщику (рис. 3.11).

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

Рис. 3.11: Используем команду ld, создавая файл main

Проверяем созданся ли исполняемый файл hello (рис. 3.12).

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

Рис. 3.12: Используем команду ls

## 3.5 Запуск исполняемого файла

Запускаем на исполнение созданный исполняемый файл (рис. 3.13).

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

Рис. 3.13: Используем команду ./hello

## 3.6 Задание для самостоятельной работы

Создаём копию файла hello.asm (рис. 3.14).

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

Рис. 3.14: Используем команду cp

Открываем файл и редактируем его (рис. 3.15 и 3.16).

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

Рис. 3.15: Открываем файл в текстовом редакторе

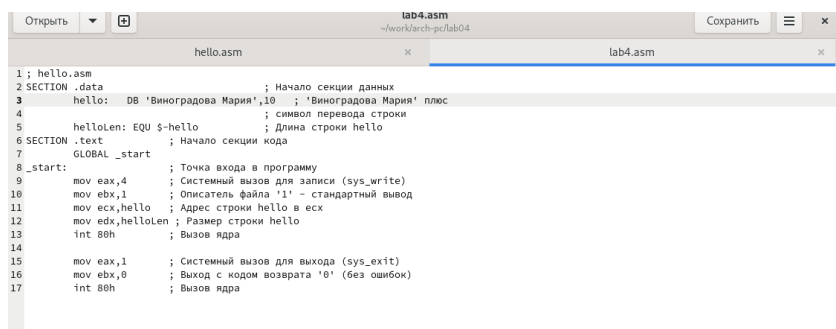


Рис. 3.16: Редактируем файл для вывода своего имени и фамилии

Прописываем те же команды, что и с первой программой (рис. 3.17).

```
mavinogradova@fedora:~/work/arch-pc/lab04$ gedit lab4.asm
mavinogradova@fedora:~/work/arch-pc/lab04$ nasm -f elf lab4.asm
mavinogradova@fedora:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.
mavinogradova@fedora:~/work/arch-pc/lab04$ ld -m elf_i386 lab4.o -o hello
mavinogradova@fedora:~/work/arch-pc/lab04$
mavinogradova@fedora:~/work/arch-pc/lab04$ ld -m elf_i386 obj.o -o main
mavinogradova@fedora:~/work/arch-pc/lab04$ ./hello
Виноградова Мария
mavinogradova@fedora:~/work/arch-pc/lab04$
```

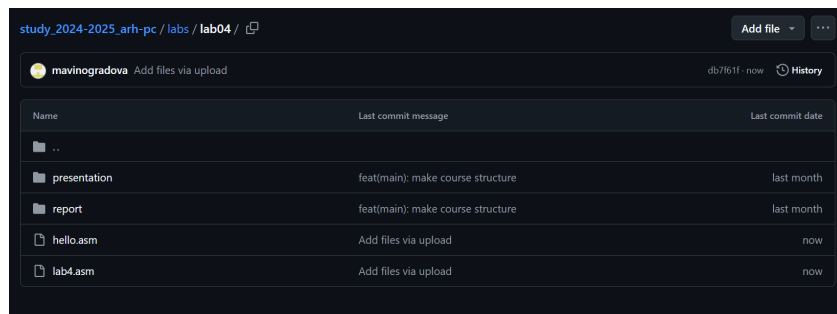
Рис. 3.17: Прописываем команды для работы файла и запускаем программу

Копируем файлы в локальный репозиторий (рис. 3.18).

```
mavinogradova@fedora: ~/work/arch-pc/lab04$ cp hello.asm ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc/labs/lab04/  
mavinogradova@fedora: ~/work/arch-pc/lab04$ cp lab4.asm ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc/labs/lab04/  
mavinogradova@fedora: ~/work/arch-pc/lab04$
```

Рис. 3.18: Копируем файлы в каталог с ЛР4

Переходим на Github и загружаем файлы (рис. 3.19).



The screenshot shows the GitHub web interface for the repository 'study\_2024-2025\_arch-pc / labs / lab04'. At the top, there is a button 'Add file' with a dropdown arrow and a three-dot menu. Below this is a section for 'mavinogradova' with the text 'Add files via upload' and a commit hash 'db7f61f' with a 'now' timestamp and a 'History' link. The main part of the interface is a table listing the repository's contents.

Name	Last commit message	Last commit date
..		
presentation	feat(main): make course structure	last month
report	feat(main): make course structure	last month
hello.asm	Add files via upload	now
lab4.asm	Add files via upload	now

Рис. 3.19: Загружаем файлы

## 4 Выводы

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