

# Шаблон отчёта по лабораторной работе №4

## Дисциплина: Архитектура компьютера

Эмаимо Елиджах Джон

## Содержание

1	Цель работы	4
2	Задание	5
3	Выполнение лабораторной работы	6
4	Выполнение заданий для самостоятельной работы	9
5	Выводы	12
6	Список литературы	13

## Список иллюстраций

3.1	Рис 1.0	6
3.2	Рис 1.1	4
3.3	Рис 1.2	4
3.4	Рис 1.3	5
3.5	Рис 2.0	5
3.6	Рис 2.1	5
3.7	Рис 3.0	5
3.8	Рис 4.0	6
3.9	Рис 4.1	6
3.10	Рис 5.0	6
4.1	Рис 6.0	6
4.2	Рис 6.1	7
4.3	Рис 6.2	7

4.4 Рис 6.3.....	7
4.5 Рис 6.4.....	7
4.6 Рис 6.5.....	8
4.7 Рис 6.6.....	8
4.8 Рис 6.7.....	8
4.9 Рис 6.8.....	8

# 1 Цель работы

Цель данной лабораторной работы - освоить 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

# 3 Выполнение лабораторной работы

## 1. Создание программы Hello world!

Создаю каталог для работы с программами на языке ассемблера NASM и перехожу в созданный каталог:

```
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ cd  
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~$ mkdir -p ~/work/arch-pc/lab04  
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~$ cd ~/work/arch-pc/lab04  
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$
```

Рис. 3.1: Рис 1.0

Создаю текстовый файл с именем hello.asm и проверяю с помощью ls:

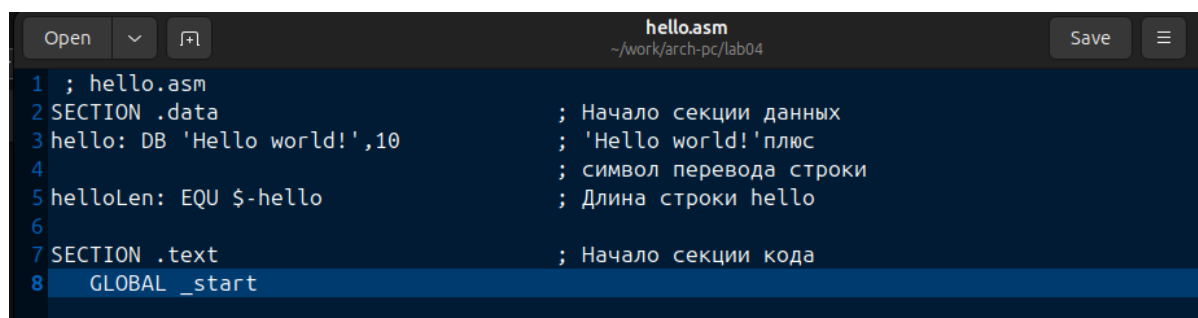
```
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ touch hello.asm  
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ ls  
hello.asm  
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$
```

Рис. 3.2: Рис 1.1

Открываю созданный файл в текстовом редакторе gedit и вставляю в него программу для вывода “Hello world!”:

```
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ gedit hello.asm
```

Рис. 3.3: Рис 1.2



```
hello.asm  
~/work/arch-pc/lab04  
1 ; hello.asm  
2 SECTION .data ; Начало секции данных  
3 hello: DB 'Hello world!',10 ; 'Hello world!'плюс  
4 ; символ перевода строки  
5 helloLen: EQU $-hello ; Длина строки hello  
6  
7 SECTION .text ; Начало секции кода  
8 GLOBAL _start
```

Рис. 3.4: Рис 1.3

## 2. Работа с транслятором NASM

Превращаю текст программы для вывода “Hello world!” в объектный код с помощью транслятора NASM, используя команду `nasm -f elf hello.asm`. Потом проверяю с помощью `ls`:

```
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ nasm -f elf hello.asm
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$
```

Рис. 3.5: Рис 2.0

```
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ ls
hello.asm hello.o
```

Рис. 3.6: Рис 2.1

## 3. Работа с расширенным синтаксисом командной строки NASM

Ввожу команду, которая скомпилирует файл `hello.asm` в файл `obj.o` и проверяю с помощью `ls`:

```
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ nasm -o obj.o -f elf
-g -l list.lst hello.asm
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ ls
hello.asm hello.o list.lst obj.o
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$
```

Рис. 3.7: Рис 3.0

## 4. Работа с компоновщиком LD

Передаю объектный файл `hello.o` на обработку компоновщику LD, чтобы получить исполняемый файл `hello` и проверяю с помощью `ls`:

```
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ ld -m elf_i386 obj.o -o hello
ld: warning: cannot find entry symbol _start; defaulting to 08049000
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o list.lst obj.o
```

Рис. 3.8: Рис 4.0

Выполняю следующую команду (рис. 4.1). Исполняемый файл будет иметь имя `main`, т.к. после ключа `-o` было задано значение `main`. Объектный файл, из которого собран этот исполняемый файл, имеет имя `obj.o`:

```
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ ld -m elf_i386 obj.o -o main
ld: warning: cannot find entry symbol _start; defaulting to 00049000
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ ls
hello  hello.asm  hello.o  list.lst  main  obj.o
```

Рис. 3.9: Рис 4.1

## 5. Запуск исполняемого файла

Запускаю на выполнение созданный файл `hello`:

```
Hello world!
```

Рис. 3.10: Рис 5.0

# 4 Выполнение заданий для самостоятельной работы

С помощью `cp` создаю копию файла `hello.asm` с именем `lab4.asm`:

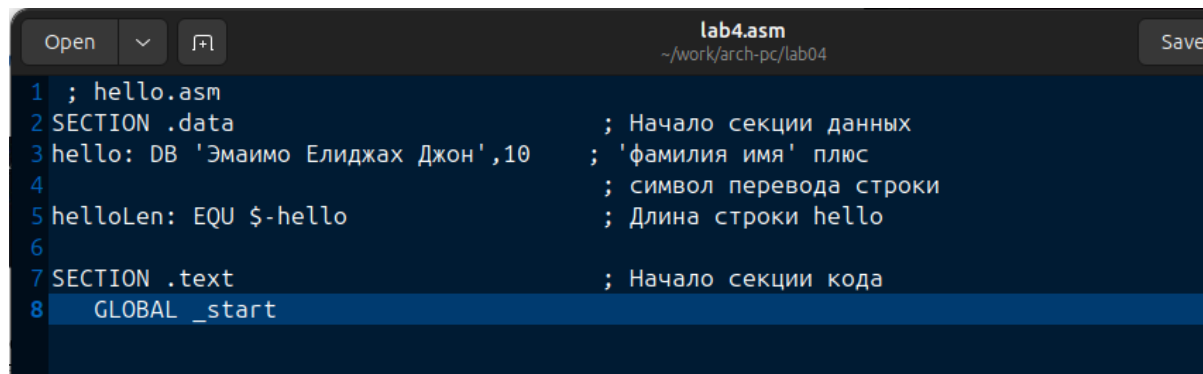
```
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ cp hello.asm lab4.asm
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ ls
hello  hello.asm  hello.o  lab4.asm  list.lst  main  obj.o
```

Рис. 4.1: Рис 6.0

С помощью `gedit` открываю файл `lab5.asm` и вношу изменения в программу так, чтобы она выводила имя и фамилию:

```
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ gedit lab4.asm
```

Рис. 4.2: Рис 6.1



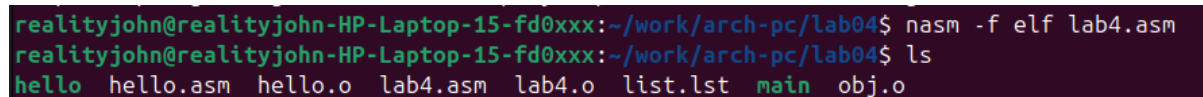
```

1 ; hello.asm
2 SECTION .data                ; Начало секции данных
3 hello: DB 'Эмаимо Елиджах Джон',10    ; 'фамилия имя' плюс
4                                     ; символ перевода строки
5 helloLen: EQU $-hello          ; Длина строки hello
6
7 SECTION .text                ; Начало секции кода
8 GLOBAL _start

```

Рис. 4.3: Рис 6.2

Компилирую текст программы в объектный файл и проверяю с помощью ls, что файл lab4.o создан:



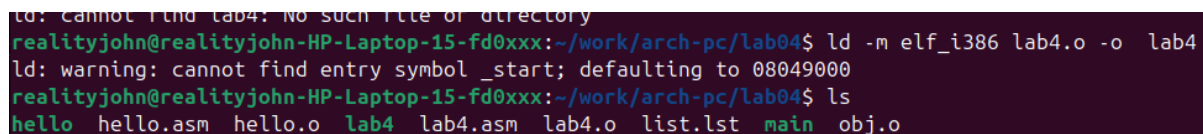
```

realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ nasm -f elf lab4.asm
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ ls
hello  hello.asm  hello.o  lab4.asm  lab4.o  list.lst  main  obj.o

```

Рис. 4.4: Рис 6.3

Передаю объектный файл lab4.o на обработку компоновщику LD, чтобы получить файл lab4:



```

ld: cannot find lab4: no such file or directory
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ ld -m elf_i386 lab4.o -o lab4
ld: warning: cannot find entry symbol _start; defaulting to 08049000
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ ls
hello  hello.asm  hello.o  lab4  lab4.asm  lab4.o  list.lst  main  obj.o

```

Рис. 4.5: Рис 6.4

Запускаю исполняемый файл lab4:



```

Эмаимо Елиджах Джон

```

Рис. 4.6: Рис 6.5

Копирую файлы hello.asm и lab4.asm в локальный репозиторий с помощью ср:

```
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ cd ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc/labs/lab04
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ ls
hello.asm lab4.asm lab4.o list.lst main obj.o presentation report
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ cp hello.asm ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc/labs/lab04
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ cp lab4.asm ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc/labs/lab04
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/arch-pc/lab04$ cd ~/work/study/2024-2025/"Архитектура компьютера"/arch-pc/labs/lab04
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$
```

Рис. 4.7: Рис 6.6

С помощью команд git add . и git commit добавляю файлы на GitHub:

```
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ git add .
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ git commit -m "Add files"
[master 1351c8e] Add files
15 files changed, 50 insertions(+)
create mode 100644 labs/lab04/hello.asm
create mode 100644 labs/lab04/lab4.asm
```

Рис. 4.8: Рис 6.7

Отправляю файлы на сервер с помощью команды git push:

```
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$ git push
Enumerating objects: 17, done.
Counting objects: 100% (17/17), done.
Delta compression using up to 12 threads
Compressing objects: 100% (13/13), done.
Writing objects: 100% (13/13), 2.29 KiB | 1.15 MiB/s, done.
Total 13 (delta 7), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (7/7), completed with 2 local objects.
To github.com:Realitize56/study_2024-2025_arch-pc.git
af07ecb..1351c8e master -> master
realityjohn@realityjohn-HP-Laptop-15-fd0xxx:~/work/study/2024-2025/Архитектура компьютера/arch-pc/labs/lab04$
```

Рис. 4.9: Рис 6.8



## 5 Выводы

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

## 6 Список литературы

Архитектура ЭВМ