

# **Отчёт по лабораторной работе №4**

**Архитектура компьютера**

Агапова Анна Антоновна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
<b>3</b>	<b>Выводы</b>	<b>9</b>
<b>4</b>	<b>Список литературы</b>	<b>10</b>

## Список иллюстраций

2.1	Создание каталога .....	6
2.2	Переход в каталог .....	6
2.3	Создание текстового файла.....	6
2.4	Открытие файла.....	6
2.5	Ввод текста .....	6
2.6	Компиляция текста .....	7
2.7	Проверка наличия объектного файла .....	7
2.8	Создание файлов.....	7
2.9	Проверка наличия файлов .....	7
2.10	Передача файла компоновщику.....	7
2.11	Проверка наличия файла hello .....	7
2.12	Задача имя исполняемого файла.....	7
2.13	Запуск на выполнение созданный исполняемый файл .....	8
2.14	Создание копии файла.....	8
2.15	Проверка создания копии файла.....	8
2.16	Оттранслирование, компоновка, запуск .....	8

## **Список таблиц**

# 1 Цель работы

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

## 2 Выполнение лабораторной работы

1. Создаю каталог для работы с программами на языке ассемблера NASM (рис. 2.1).

```
aaagapova@astra:~$ mkdir -p ~/work/arch-pc/lab04
```

Рисунок 2.1: Создание каталога

2. Перехожу в созданный каталог (рис. 2.2).

```
aaagapova@astra:~$ cd ~/work/arch-pc/lab04
```

Рисунок 2.2: Переход в каталог

3. Создаю текстовый файл с именем `hello.asm` (рис. 2.3).

```
aaagapova@astra:~/work/arch-pc/lab04$ touch hello.asm
```

Рисунок 2.3: Создание текстового файла

4. Открою этот файл с помощью текстового редактора Kate (рис. 2.4).

```
aaagapova@astra:~/work/arch-pc/lab04$ kate hello.asm
```

Рисунок 2.4: Открытие файла

5. Ввожу в него следующий текст (рис. 2.5)

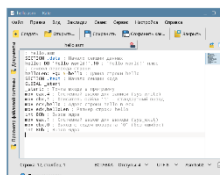


Рисунок 2.5: Ввод текста

6. Скомпилируем данный текст (рис. 2.6).

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

Рисунок 2.6: Компиляция текста

7. Проверю, что объектный файл был создан (рис. 2.7).

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

Рисунок 2.7: Проверка наличия объектного файла

8. Скомпилирую исходный файл hello.asm в obj.o и создадим файл листинга list.lst (рис. 2.8).

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

Рисунок 2.8: Создание файлов

9. Проверю, что файлы были созданы (рис. 2.9).

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

Рисунок 2.9: Проверка наличия файлов

10. Передаю объектный файл на обработку компоновщику (рис. 2.10).

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

Рисунок 2.10: Передача файла компоновщику

11. Проверю, что исполняемый файл hello был создан (рис. 2.11).

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

Рисунок 2.11: Проверка наличия файла hello

12. Задам имя создаваемого исполняемого файла (рис. 2.12).

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

Рисунок 2.12: Задача имя исполняемого файла

13. Запущу на выполнение созданный исполняемый файл, находящийся в текущем каталоге (рис. 2.13).

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

Рисунок 2.13: Запуск на выполнение созданный исполняемый файл

14. Создам копию файла `hello.asm` с именем `lab04.asm` (рис. 2.14).

```
aaagapova@astra:~/work/arch-pc/lab04$ cp hello.asm lab04.asm
```

Рисунок 2.14: Создание копии файла

15. Проверяю, что копия создалась с именем `lab04.asm` (рис. 2.15).

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

Рисунок 2.15: Проверка создании копии файла

16. Оттранслирую полученный текст программы `lab04.asm` в объектный файл. Выполню компоновку объектного файла и запущу получившийся исполняемый файл (рис. 2.16).

```
aaagapova@astra:~/work/arch-pc/lab04$ kate lab04.asm
aaagapova@astra:~/work/arch-pc/lab04$ nasm -f elf lab04.asm
aaagapova@astra:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.lst lab04.asm
aaagapova@astra:~/work/arch-pc/lab04$ ld -m elf_i386 obj.o -o lab04
aaagapova@astra:~/work/arch-pc/lab04$ ld -m elf_i386 lab04.o -o lab04
aaagapova@astra:~/work/arch-pc/lab04$ ./lab04
Ragova Anna!
```

Рисунок 2.16: Оттранслирование, компоновка, запуск

17. Скопирую файлы `hello.asm` и `lab04.asm` в локальный репозиторий и загружу файлы на Github.



## **3 Выводы**

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

## 4 Список литературы

1. GDB: The GNU Project Debugger. — URL: <https://www.gnu.org/software/gdb/>.
2. GNU Bash Manual. — 2016. — URL: <https://www.gnu.org/software/bash/manual/>.
3. Midnight Commander Development Center. — 2021. — URL: <https://midnight-commander.org/>.
4. NASM Assembly Language Tutorials. — 2021. — URL: <https://asmtutor.com/>.
5. 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>.
6. Robbins A. Bash Pocket Reference. — O'Reilly Media, 2016. — 156 с. — ISBN 978-1491941591.
7. The NASM documentation. — 2021. — URL: <https://www.nasm.us/docs.php>.
8. Zarrelli G. Mastering Bash. — Packt Publishing, 2017. — 502 с. — ISBN 9781784396879.
9. Колдаев В. Д., Лупин С. А. Архитектура ЭВМ. — М. : Форум, 2018.
10. Куляс О. Л., Никитин К. А. Курс программирования на ASSEMBLER. — М. : Солон-Пресс, 2017.
11. Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
12. Расширенный ассемблер: NASM. — 2021. — URL: <https://www.opennet.ru/docs/RUS/nasm/>.
13. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВПетербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.
14. 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](http://www.stolyarov.info/books/asm_unix).
15. Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. — 874 с. — (Классика Computer Science).
16. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер, 2015. — 1120 с. — (Классика Computer Science).