

# **Отчёт по лабораторной работе №4**

**Дисциплина: архитектура компьютера**

Лысенко Маргарита Олеговна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
<b>5</b>	<b>Выполнение самостоятельной работы</b>	<b>10</b>
<b>6</b>	<b>Выводы</b>	<b>12</b>

# Список иллюстраций

4.1	Создание каталога и переход в него . . . . .	8
4.2	Создание файла . . . . .	8
4.3	Компиляция текста . . . . .	9
4.4	Компиляция исходного файла и проверка . . . . .	9
4.5	Передача файла компоновщику и проверка . . . . .	9
4.6	Присвоение имени . . . . .	9
4.7	Запуск файла . . . . .	9
5.1	Копирование и редактирование файла . . . . .	10
5.2	Компоновка и запуск объектного файла . . . . .	11
5.3	Загрузка файлов на Github . . . . .	11

## **Список таблиц**

# 1 Цель работы

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

## 2 Задание

Вывести программы Hello world и своё имя, фамилию.

### 3 Теоретическое введение

Основными функциональными элементами любой электронно-вычислительной машины (ЭВМ) являются центральный процессор, память и периферийные устройства. Взаимодействие этих устройств осуществляется через общую шину, к которой они подключены. Физически шина представляет собой большое количество проводников, соединяющих устройства друг с другом. В современных компьютерах проводники выполнены в виде электропроводящих дорожек на материнской (системной) плате. Основной задачей процессора является обработка информации, а также организация координации всех узлов компьютера. В состав центрального процессора (ЦП) входят следующие устройства: • арифметико-логическое устройство (АЛУ) — выполняет логические и арифметические действия, необходимые для обработки информации, хранящейся в памяти; • устройство управления (УУ) — обеспечивает управление и контроль всех устройств компьютера; • регистры — сверхбыстрая оперативная память небольшого объёма, входящая в состав процессора, для временного хранения промежуточных результатов выполнения инструкций; регистры процессора делятся на два типа: регистры общего назначения и специальные регистры.

## 4 Выполнение лабораторной работы

Создала каталог для работы с программами на языке ассемблера NASM (рис. 4.1).

```
hsenko@dk8n64 ~ $ mkdir -p ~/work/arch-pc/lab04  
hsenko@dk8n64 ~ $ cd ~/work/arch-pc/lab04
```

Рис. 4.1: Создание каталога и переход в него

Создала текстовый файл с именем hello.asm (рис. 4.2).

```
hsenko@dk8n64 ~/work/arch-pc/lab04 $ touch hello.asm
```

Рис. 4.2: Создание файла

Листинг программы Hello, world:

```
SECTION .data hello: db "Hello, world!",0xa helloLen: equ $ - hello  
SECTION .text  
global _start  
_start:  
mov eax, 4  
mov ebx, 1  
mov ecx, hello mov edx, helloLen int 0x80  
  
mov eax, 1  
mov ebx, 0  
int 0x80
```



Скомпилировала текст программы «Hello World» (рис. 4.3).

```
molihsenko@dk8n64 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
```

Рис. 4.3: Компиляция текста

Скомпилировала исходный файл hello.asm в obj.o и сделала проверку (рис. 4.4).

```
molihsenko@dk8n64 ~/work/arch-pc/lab04 $ nasm -o obj.o -f elf -g -l list.lst hello.asm
molihsenko@dk8n64 ~/work/arch-pc/lab04 $ ls
hello.asm  hello.o  list.lst  obj.o
```

Рис. 4.4: Компиляция исходного файла и проверка

Передала объектный файл на обработку компоновщику и сделала проверку (рис. 4.5).

```
molihsenko@dk8n64 ~/work/arch-pc/lab04 $ ld -m elf_i386 hello.o -o hello
molihsenko@dk8n64 ~/work/arch-pc/lab04 $ ls
hello  hello.asm  hello.o  list.lst  obj.o
```

Рис. 4.5: Передача файла компоновщику и проверка

С помощью ключа -o с последующим значением задала в данном случае имя создаваемого исполняемого файла. (рис. 4.6).

```
molihsenko@dk8n64 ~/work/arch-pc/lab04 $ ld -m elf_i386 obj.o -o main
```

Рис. 4.6: Присвоение имени

Запустила на выполнение созданный исполняемый файл, находящийся в текущем каталоге (рис. 4.7).

```
molihsenko@dk8n64 ~/work/arch-pc/lab04 $ ./hello
Hello, world!
```

Рис. 4.7: Запуск файла

## 5 Выполнение самостоятельной работы

С помощью команды `cp` создала копию файла `hello.asm` с именем `lab4.asm`. Внесла изменения в текст программы в файле `lab4.asm` так, чтобы вместо `Hello world!` на экран выводилась строка с моими фамилией и именем. (рис. 5.1).

```
molihsenko@dk8n64 ~/work/arch-pc/lab04 $ cp hello.asm lab4.asm
molihsenko@dk8n64 ~/work/arch-pc/lab04 $ gedit lab4.asm
molihsenko@dk8n64 ~/work/arch-pc/lab04 $ nasm -f elf lab4.asm
molihsenko@dk8n64 ~/work/arch-pc/lab04 $ nasm -o obj.o -f elf -g -l list.lst lab4.asm
```

Рис. 5.1: Копирование и редактирование файла

Листинг программы Лысенко Маргарита:

```
SECTION .data hello: db "Лысенко Маргаита",0xa helloLen: equ $ - hello
SECTION .text global _start
_start:
mov eax, 4
mov ebx, 1
mov ecx, hello mov edx, helloLen int 0x80

mov eax, 1
    mov ebx, 0
    int 0x80
```

Оттранслировала полученный текст программы `lab4.asm` в объектный файл. Выполнила компоновку объектного файла и запустила получившийся исполняемый файл (рис. 5.2).

```

molihsenko@dk8n81 ~/work/arch-pc/lab04 $ ld -m elf_i386 lab4.o -o hello
molihsenko@dk8n81 ~/work/arch-pc/lab04 $ ld -m elf_i386 obj.o -o main
molihsenko@dk8n81 ~/work/arch-pc/lab04 $ ./hello
Лысенко Маргарита

```

Рис. 5.2: Компоновка и запуск объектного файла

Скопировала файлы hello.asm и lab4.asm в свой локальный репозиторий в каталог ~/work/study/2023-2024/“Архитектура компьютера”/arch-pc/labs/lab04/. Загрузила файлы на Github. (рис. 5.3).

```

molihsenko@dk8n81 ~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab04 $ git push origin
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 6 потоков
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 677 байтов | 677.00 КиБ/с, готово.
Всего 6 (изменений 3), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (3/3), completed with 2 local objects.
To github.com:LysenkoM/arch-ps.git
   e6e02eb..f79b188  master -> master

```

Рис. 5.3: Загрузка файлов на Github

## 6 Выводы

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