

# **Отчет по лабораторной работе №4**

**Архитектура компьютера**

**Элсаиед Адел**

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
2.0.1	1 . . . . .	6
2.0.2	2 . . . . .	6
2.0.3	3 . . . . .	6
2.0.4	4 . . . . .	7
2.0.5	5 . . . . .	7
2.0.6	6 . . . . .	8
2.0.7	7 . . . . .	8
2.0.8	8 . . . . .	8
2.0.9	9 . . . . .	9
<b>3</b>	<b>Самостоятельная работа</b>	<b>10</b>
3.0.1	1 . . . . .	10
3.0.2	2 . . . . .	10
3.0.3	3 . . . . .	11
3.0.4	4 . . . . .	11
<b>4</b>	<b>Выводы</b>	<b>13</b>

## Список иллюстраций

2.1	Создание каталога с помощью команд <code>mkdir -p ~/work/arch-pc/lab04</code>	6
2.2	Переход в созданный каталог с помощью команд <code>cd ~/work/arch-pc/lab04</code>	6
2.3	Создание текстового файла с помощью команд <code>touch hello.asm</code>	7
2.4	Открытие текстового редактора <code>gedit</code> с помощью команды <code>gedit hello.asm</code>	7
2.5	И ввожу в него следующий текст.	7
2.6	Ввожу команду <code>nasm -f elf hello.asm</code>	8
2.7	Расширенный синтаксис командной строки NASM.	8
2.8	Компоновщик LD.	8
2.9	Ввожу команду <code>ld -m elf_i386 obj.o -o main</code>	8
2.10	Ввожу команду <code>./hello</code>	9
3.1	Создаю копию файла <code>hello.asm</code> с именем <code>lab04.asm</code>	10
3.2	Ввожу свое имя фамилию.	10
3.3	Запускаю получившийся исполняемый файл.	11
3.4	Копирую файлы <code>hello.asm</code> и <code>lab4.asm</code> с помощью команды <code>cp hello.asm lab04.asm ~/work/study/2023-2024/“Архитектура компьютера”/arch-pc/labs/lab04/</code>	11
3.5	Проверяю.	11
3.6	Загружаю файлы на Github.	12

## Список таблиц

# 1 Цель работы

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

## 2 Выполнение лабораторной работы

### 2.0.1 1

Создаю каталог для работы с программами на языке ассемблера NASM.

```
[aaehsaied@fedora arch-pc]$ mkdir -p ~/work/arch-pc/lab04  
[aaehsaied@fedora arch-pc]$
```

Рис. 2.1: Создание каталога с помощью команд `mkdir -p ~/work/arch-pc/lab04`

### 2.0.2 2

Перехожу в созданный каталог.

```
[aaehsaied@fedora arch-pc]$ cd ~/work/arch-pc/lab04  
[aaehsaied@fedora lab04]$
```

Рис. 2.2: Переход в созданный каталог с помощью команд `cd ~/work/arch-pc/lab04`

### 2.0.3 3

Создаю текстовый файл с именем `hello.asm`

```
[aaehsaied@fedora lab04]$ touch hello.asm
```

Рис. 2.3: Создание текстового файла с помощью команд touch hello.asm

## 2.0.4 4

Открываю этот файл с помощью текстового редактора gedit.

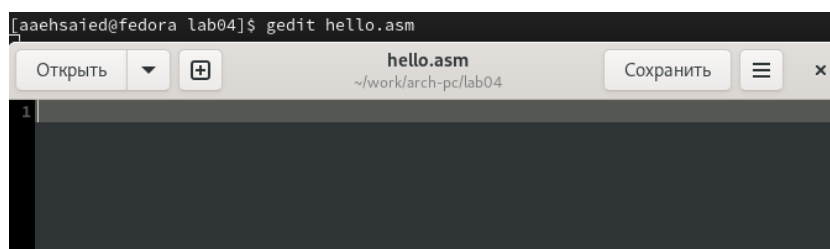


Рис. 2.4: Открытие текстового редактора gedit с помощью команды gedit hello.asm

A screenshot of the gedit text editor window showing assembly code. The title bar is the same as in the previous image. The code is as follows:

```
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 SECTION .text ; Начало секции кода
7 GLOBAL _start
8 _start: ; Точка входа в программу
9 mov eax,4 ; Системный вызов для записи (sys_write)
10 mov ebx,1 ; Описатель файла '1' - стандартный вывод
11 mov ecx,hello ; Адрес строки hello в ecx
12 mov edx,helloLen ; Размер строки hello
13 int 80h ; Вызов ядра
14 mov eax,1 ; Системный вызов для выхода (sys_exit)
15 mov ebx,0 ; Выход с кодом возврата '0' (без ошибок)
16 int 80h ; Вызов ядра
```

Рис. 2.5: И ввожу в него следующий текст.

## 2.0.5 5

NASM превращает текст программы в объектный код.

```
[aaehsaied@fedora lab04]$ nasm -f elf hello.asm  
[aaehsaied@fedora lab04]$
```

Рис. 2.6: Ввожу команду `nasm -f elf hello.asm`

## 2.0.6 6

Полный вариант командной строки `nasm` выглядит следующим образом:

```
[aaehsaied@fedora lab04]$ nasm -o obj.o -f elf -g -l list.lst hello.asm  
[aaehsaied@fedora lab04]$
```

Рис. 2.7: Расширенный синтаксис командной строки `NASM`.

## 2.0.7 7

Чтобы получить исполняемую программу, объектный файл необходимо передать на обработку компоновщику:

```
[aaehsaied@fedora lab04]$ ld -m elf_i386 hello.o -o hello  
[aaehsaied@fedora lab04]$
```

Рис. 2.8: Компоновщик `LD`.

## 2.0.8 8

Ключ `-o` с последующим значением задаёт в данном случае имя создаваемого исполняемого файла.

```
[aaehsaied@fedora lab04]$ ld -m elf_i386 obj.o -o main  
[aaehsaied@fedora lab04]$
```

Рис. 2.9: Ввожу команду `ld -m elf_i386 obj.o -o main`



## 2.0.9 9

Запуск исполняемого файла.

```
[aaehsaied@fedora lab04]$ ./hello  
Hello world!  
[aaehsaied@fedora lab04]$
```

Рис. 2.10: Ввожу команду ./hello

## 3 Самостоятельная работа

### 3.0.1 1

В каталоге `~/work/arch-pc/lab04` с помощью команды `cp`

```
[aaehsaied@fedora lab04]$ cp hello.asm lab04.asm  
[aaehsaied@fedora lab04]$
```

Рис. 3.1: Создаю копию файла `hello.asm` с именем `lab04.asm`

### 3.0.2 2

С помощью текстового редактора `gedit` ввожу изменения в тексте программы в файле `lab04.asm` вместо `Hello world!` ввожу Бердыев Даянч.

```
[aaehsaied@fedora lab04]$ gedit lab04.asm  
Открыть ▼ + lab04.asm Сохранить ☰ x  
~/work/arch-pc/lab04  
1; hello.asm  
2SECTION .data ; Начало секции данных  
3hello: DB 'Элсаиед Адел!',10 ; 'Элсаиед Адел!' плюс  
4; символ перевода строки  
5helloLen: EQU $-hello ; Длина строки hello  
6SECTION .text ; Начало секции кода  
7GLOBAL _start  
8_start: ; Точка входа в программу  
9mov eax,4 ; Системный вызов для записи (sys_write)  
10mov ebx,1 ; Описатель файла '1' - стандартный вывод  
11mov ecx,hello ; Адрес строки hello в ecx  
12mov edx,helloLen ; Размер строки hello  
13int 80h ; Вызов ядра  
14mov eax,1 ; Системный вызов для выхода (sys_exit)  
15mov ebx,0 ; Выход с кодом возврата '0' (без ошибок)  
16int 80h ; Вызов ядра
```

Рис. 3.2: Ввожу свое имя фамилию.

### 3.0.3 3

Оттранслирую полученный текст программы lab04.asm в объектный файл. Выполняю компоновку объектного файла.

```
[aaehsaied@fedora lab04]$ nasm -f elf lab04.asm
[aaehsaied@fedora lab04]$ nasm -o obj.o -f elf -g -l list.lst lab04.asm
[aaehsaied@fedora lab04]$ ld -m elf_i386 lab04.o -o lab04
[aaehsaied@fedora lab04]$ ld -m elf_i386 obj.o -o main
[aaehsaied@fedora lab04]$ ./lab04
Элсаиед Адел!
[aaehsaied@fedora lab04]$
```

Рис. 3.3: Запускаю получившийся исполняемый файл.

### 3.0.4 4

Копирую файлы hello.asm и lab04.asm в локальный репозиторий в каталог ~/work/study/2023-2024/“Архитектура компьютера”/arch-pc/labs/lab04/.

```
[aaehsaied@fedora lab04]$ cp hello.asm lab04.asm ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab04/
[aaehsaied@fedora lab04]$
```

Рис. 3.4: Копирую файлы hello.asm и lab4.asm с помощью команды cp hello.asm lab04.asm ~/work/study/2023-2024/“Архитектура компьютера”/arch-pc/labs/lab04/

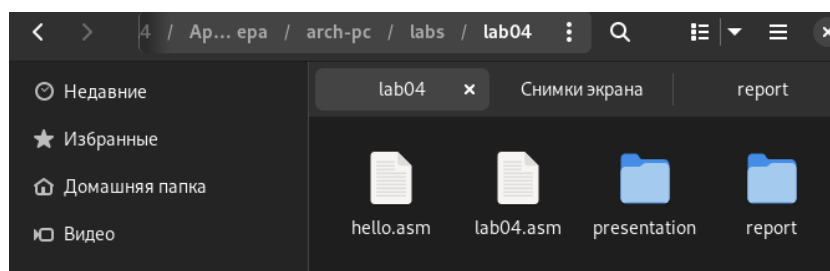
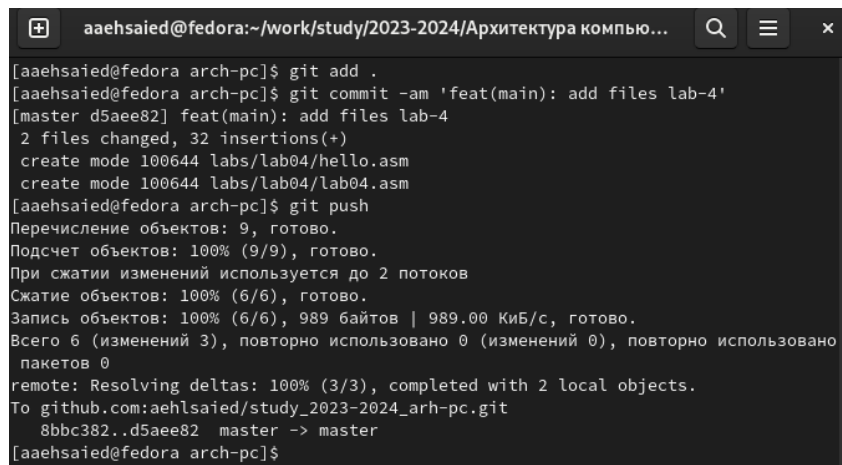


Рис. 3.5: Проверяю.



```
aaehsaied@fedora:~/work/study/2023-2024/Архитектура компью...
[aaehsaied@fedora arch-pc]$ git add .
[aaehsaied@fedora arch-pc]$ git commit -am 'feat(main): add files lab-4'
[master d5aee82] feat(main): add files lab-4
2 files changed, 32 insertions(+)
create mode 100644 labs/lab04/hello.asm
create mode 100644 labs/lab04/lab04.asm
[aaehsaied@fedora arch-pc]$ git push
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 2 потоков
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 989 байтов | 989.00 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:aehtsaied/study_2023-2024_arh-pc.git
8bbc382..d5aee82 master -> master
[aaehsaied@fedora arch-pc]$
```

Рис. 3.6: Загружаю файлы на Github.

## 4 Выводы

В ходе выполнения этой лабораторной работы я освоил процедуру компиляции и сборки программ, написанных на ассемблере NASM.