

# **Отчёт по лабораторной работе №4**

**Дисциплина: Архитектура компьютера**

Толстых Александра Андреевна

# 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	Компоновщик LD . . . . .	8
3.5	Запуск исполняемого файла . . . . .	9
3.6	Задание для самостоятельной работы . . . . .	9
<b>4</b>	<b>Выводы</b>	<b>12</b>

# Список иллюстраций

3.1	Создание и редактирование hello.asm . . . . .	7
3.2	Введение текста . . . . .	7
3.3	Компиляция текста программы . . . . .	8
3.4	Компиляция текста программы . . . . .	8
3.5	Передача объектного файла на обработку компоновщику . . . . .	8
3.6	Передача объектного файла на обработку компоновщику . . . . .	9
3.7	Запуск исполняемого файла . . . . .	9
3.8	Создание копии . . . . .	9
3.9	Изменение программы . . . . .	10
3.10	Компиляция текста программы . . . . .	10
3.11	Передача объектного файла на обработку компоновщику . . . . .	10
3.12	Запуск файла . . . . .	10
3.13	Создание копий файлов . . . . .	11
3.14	Сохранение изменений на гитхабе . . . . .	11

## **Список таблиц**

# 1 Цель работы

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

## 2 Задание

1. Программа Hello world!
2. Транслятор NASM
3. Расширенный синтаксис командной строки NASM
4. Компоновщик LD
5. Запуск исполняемого файла
6. Задание для самостоятельной работы

## 3 Выполнение лабораторной работы

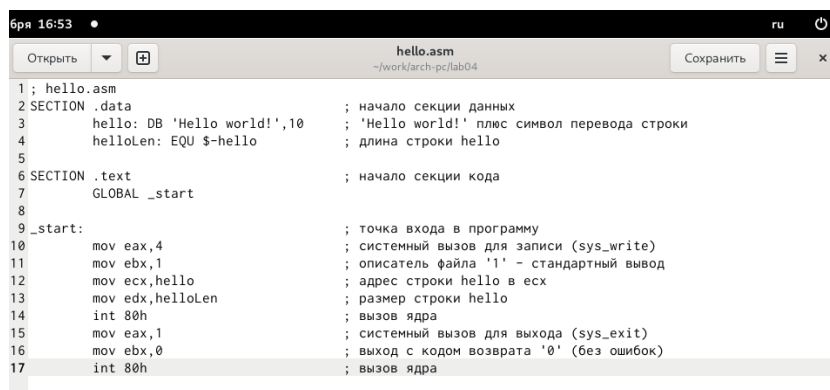
### 3.1 Программа Hello world!

Создаю каталог для работы с программами на языке ассемблера NASM. Затем перехожу в созданный каталог и создаю текстовый файл с именем hello.asm. Открываю его при помощи текстового редактора gedit. (рис. 3.1).

```
aatolstihkh@dk3n55 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc $ mkdir -p ~/work
/arch-pc/lab04
aatolstihkh@dk3n55 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc $ cd ~/work/arch-
pc/lab04
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ touch hello.asm
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ gedit hello.asm
```

Рис. 3.1: Создание и редактирование hello.asm

Ввожу указанный текст. (рис. 3.2).



The screenshot shows a gedit window titled 'hello.asm' with the following assembly code:

```
1 ; hello.asm
2 SECTION .data ; начало секции данных
3     hello: DB 'Hello world!',10 ; 'Hello world!' плюс символ перевода строки
4     helloLen: EQU $-hello ; длина строки hello
5
6 SECTION .text ; начало секции кода
7     GLOBAL _start
8
9 _start: ; точка входа в программу
10     mov eax,4 ; системный вызов для записи (sys_write)
11     mov ebx,1 ; дескриптор файла '1' - стандартный вывод
12     mov ecx,hello ; адрес строки hello в ecx
13     mov edx,helloLen ; размер строки hello
14     int 0x80 ; вызов ядра
15     mov eax,1 ; системный вызов для выхода (sys_exit)
16     mov ebx,0 ; выход с кодом возврата '0' (без ошибок)
17     int 0x80 ; вызов ядра
```

Рис. 3.2: Введение текста

## 3.2 Транслятор NASM

Превращаю текст для программы в объектный код с помощью транслятора NASM. Далее проверяю правильность выполнения команды при помощи ls. (рис. 3.3).

```
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ nasm -f elf hello.asm
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ ls
hello.asm  hello.o
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $
```

Рис. 3.3: Компиляция текста программы

Объектный файл имеет имя “hello.o”

## 3.3 Расширенный синтаксис командной строки NASM

Ввожу команду, которая скомпилирует файл hello.asm в файл obj.o, при этом в файл будут включены символы для отладки, а также будет создан файл листинга. При помощи ls также проверяю корректность выполнения программы. (рис. 3.4).

```
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ nasm -o obj.o -f elf -g -l list.lst hello.asm
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ ls
hello.asm  hello.o  list.lst  obj.o
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $
```

Рис. 3.4: Компиляция текста программы

## 3.4 Компоновщик LD

Передаю объектный файл на обработку компоновщику LD, чтобы получить исполняемый файл hello. Далее проверяю с помощью ls правильность выполнения команды. (рис. 3.5).

```
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ ld -m elf_i386 hello.o -o hello
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ ls
hello  hello.asm  hello.o  list.lst  obj.o
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $
```

Рис. 3.5: Передача объектного файла на обработку компоновщику



Выполняю следующую команду. Исполняемый файл будет иметь имя main, так как после ключа -o было задано значение main. Объектный файл, из которого собран этот исполняемый файл имеет имя obj.o. (рис. 3.6).

```
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ ld -m elf_i386 obj.o -o main
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ ls
hello hello.asm hello.o list.lst main obj.o
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $
```

Рис. 3.6: Передача объектного файла на обработку компоновщику

## 3.5 Запуск исполняемого файла

Запускаю исполняемый файл hello. (рис. 3.7).

```
hello hello.asm hello.o list.lst main obj.o
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ ./hello
Hello world!
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $
```

Рис. 3.7: Запуск исполняемого файла

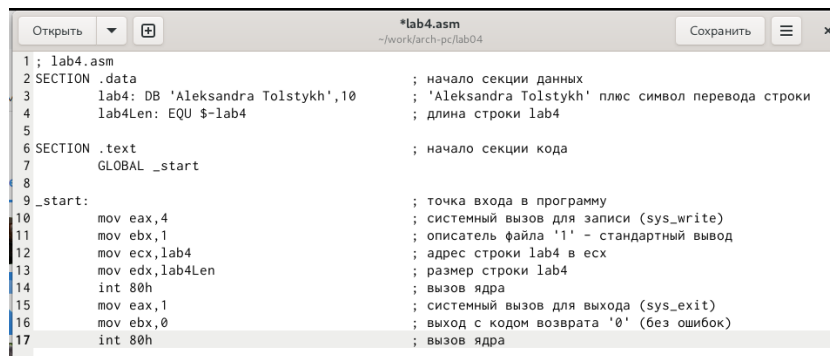
## 3.6 Задание для самостоятельной работы

В каталоге ~/work/arch-pc/lab04 с помощью команды cp создаю копию файла hello.asm с именем lab4.asm. (рис. 3.8).

```
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ cp hello.asm lab4.asm
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ gedit lab4.asm
]
```

Рис. 3.8: Создание копии

С помощью gedit вношу изменения в текст программы в файле lab4.asm так, чтобы вместо Hello world! на экран выводилась строка с моими фамилией и именем (рис. 3.9).



```
1 ; lab4.asm
2 SECTION .data
3     lab4: DB 'Aleksandra Tolstykh',10      ; начало секции данных
4     lab4Len: EQU $-lab4                  ; 'Aleksandra Tolstykh' плюс символ перевода строки
5                                         ; длина строки lab4
6 SECTION .text
7     GLOBAL _start                        ; начало секции кода
8
9 _start:
10    mov eax,4                            ; точка входа в программу
11    mov ebx,1                            ; системный вызов для записи (sys_write)
12    mov ecx,lab4                         ; описатель файла '1' - стандартный вывод
13    mov edx,lab4Len                      ; адрес строки lab4 в ecx
14    int 80h                             ; размер строки lab4
15    mov eax,1                            ; вызов ядра
16    mov ebx,0                            ; системный вызов для выхода (sys_exit)
17    int 80h                             ; выход с кодом возврата '0' (без ошибок)
18                                         ; вызов ядра
```

Рис. 3.9: Изменение программы

Транслирую полученный текст программы lab4.asm в объектный файл.(рис. 3.10).

```
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ nasm -f elf lab4.asm
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ ls
hello hello.asm hello.o lab05.asm lab4.asm lab4.o list.lst main obj.o
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $
```

Рис. 3.10: Компиляция текста программы

Выполняю компоновку объектного файла (рис. 3.11).

```
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ ld -m elf_i386 lab4.o -o lab4
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ ls
hello hello.asm hello.o lab05.asm lab4 lab4.asm lab4.o list.lst main obj.o
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $
```

Рис. 3.11: Передача объектного файла на обработку компоновщику

Запускаю получившийся исполняемый файл.(рис. 3.12).

```
hello hello.asm hello.o lab05.asm lab4 lab4.asm lab4.o list.lst main obj.o
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $ ./lab4
Aleksandra Tolstykh
aatolstihkh@dk3n55 ~/work/arch-pc/lab04 $
```

Рис. 3.12: Запуск файла

Копирую указанные файлы в локальный репозиторий в каталог с 4 лабораторной. (рис. 3.13).

```

aato1stihkh@dk3n55 ~/work/arch-pc/lab04 $ cp hello.asm ~/work/study/2024-2025/"Архитектура компьютера"/s
tudy_2024-2025_arhpc/labs/lab04
aato1stihkh@dk3n55 ~/work/arch-pc/lab04 $ cp lab4.asm ~/work/study/2024-2025/"Архитектура компьютера"/st
udy_2024-2025_arhpc/labs/lab04
aato1stihkh@dk3n55 ~/work/arch-pc/lab04 $ cd ~/work/study/2024-2025/"Архитектура компьютера"/study_2024-
2025_arhpc/labs/lab04
aato1stihkh@dk3n55 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc/labs/lab04 $ ls
hello.asm lab4.asm presentation report
aato1stihkh@dk3n55 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc/labs/lab04 $

```

Рис. 3.13: Создание копий файлов

Отправляю изменения на гитхаб. (рис. 3.14).

```

aato1stihkh@dk3n55 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc $ git add .
aato1stihkh@dk3n55 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc $ git commit -am
'feat(main): add files lab-4'
[master 1033f6a] feat(main): add files lab-4
2 files changed, 34 insertions(+)
create mode 100644 labs/lab04/hello.asm
create mode 100644 labs/lab04/lab4.asm
aato1stihkh@dk3n55 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc $ git push
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), 1.10 КиБ | 1.10 МиБ/с, готово.
Total 6 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 2 local objects.
To github.com:1111isonok/study_2024-2025_arhpc.git
2907c89..1033f6a master -> master
aato1stihkh@dk3n55 ~/work/study/2024-2025/Архитектура компьютера/study_2024-2025_arhpc $

```

Рис. 3.14: Сохранение изменений на гитхабе

## 4 Выводы

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