

Отчёт по лабораторной работе 4

Создание и процесс обработки программ на языке ассемблера NASM

Бабуджян Тигран НБИбд-01-24

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
3.1	Программа Hello world!	7
3.2	Транслятор NASM	8
3.3	Компоновщик LD	9
3.4	Выполнение заданий для самостоятельной работы.	10
4	Выводы	12
	Список литературы	13

Список иллюстраций

3.1	Создание каталога и файла	7
3.2	Программа hello.asm	8
3.3	Трансляция hello.asm	8
3.4	Трансляция hello.asm с дополнительными опциями	9
3.5	Линковка программы	9
3.6	Линковка программы	9
3.7	Запуск программ	9
3.8	Код программы в файле lab4.asm	10
3.9	Запуск программы lab4.asm	11

Список таблиц

1 Цель работы

Целью работы является освоение процедуры компиляции и сборки программ, написанных на ассемблере NASM.

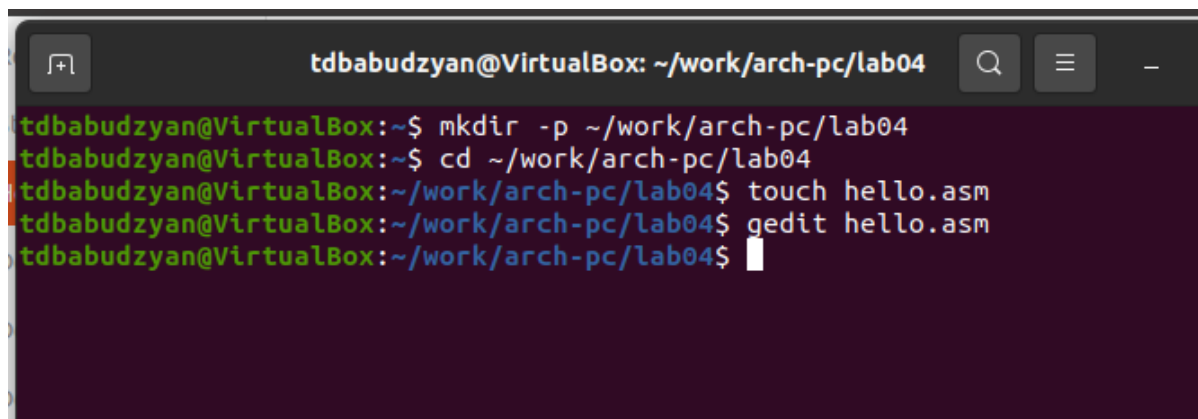
2 Задание

1. Изучите программу HelloWorld и скомпилируйте ее.
2. С помощью любого текстового редактора внесите изменения в текст программы так, чтобы вместо Hello world! на экран выводилась строка с вашими фамилией и именем.
3. Скомпилируйте новую программу и проверьте ее работу.
4. Загрузите файлы на GitHub.

3 Выполнение лабораторной работы

3.1 Программа Hello world!

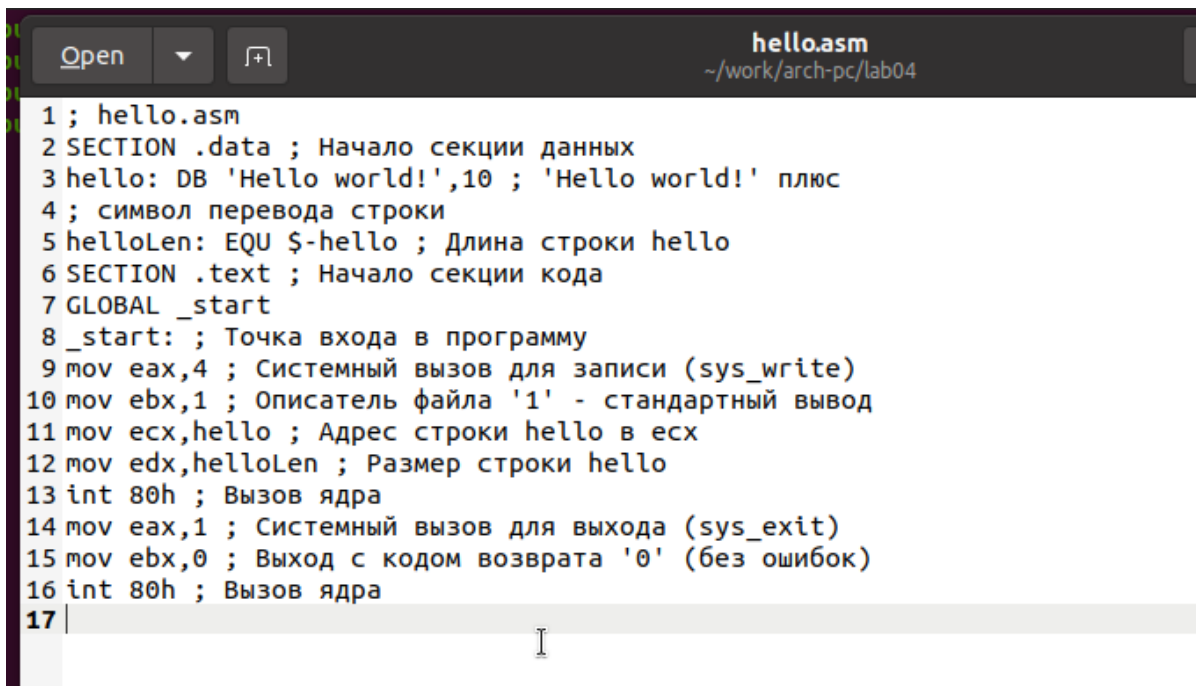
Создаю каталог lab04 командой `mkdir`, перехожу в него с помощью команды `cd`, создаю файл `hello.asm`. (рис. 3.1)

A screenshot of a terminal window with a dark purple background. The window title is "tdbabudzyan@VirtualBox: ~/work/arch-pc/lab04". The terminal shows the following commands and their outputs:

```
tdbabudzyan@VirtualBox:~$ mkdir -p ~/work/arch-pc/lab04
tdbabudzyan@VirtualBox:~$ cd ~/work/arch-pc/lab04
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$ touch hello.asm
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$ gedit hello.asm
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$
```

Рис. 3.1: Создание каталога и файла

Открыла файл и написала код программы по заданию.(рис. 3.2)

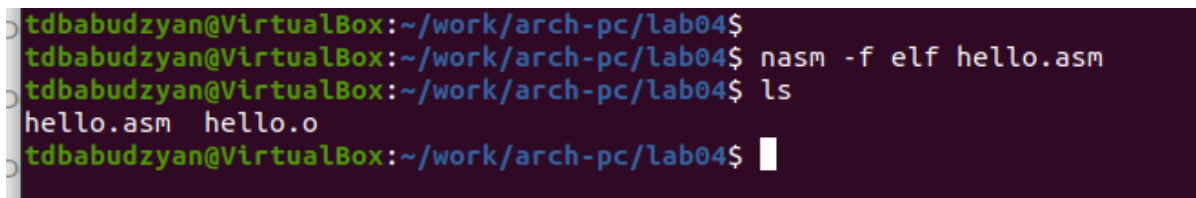


```
1 ; hello.asm
2 SECTION .data ; Начало секции данных
3 hello: DB 'Hello world!',10 ; 'Hello world!' плюс
4 ; символ перевода строки
5 helloLen: EQU $-hello ; Длина строки hello
6 SECTION .text ; Начало секции кода
7 GLOBAL _start
8 _start: ; Точка входа в программу
9 mov eax,4 ; Системный вызов для 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
17 |
```

Рис. 3.2: Программа hello.asm

3.2 Транслятор NASM

Транслирую файл командой `nasm`. Получился объектный файл `hello.o` (рис. 3.3)



```
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$ nasm -f elf hello.asm
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$ ls
hello.asm  hello.o
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$
```

Рис. 3.3: Трансляция hello.asm

Транслирую файл командой `nasm` с дополнительными опциями. (рис. 3.4)
Получился файл листинга `list.lst`, объектный файл `obj.o`, в программу добавилась отладочная информация.


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

Рис. 3.4: Трансляция hello.asm с дополнительными опциями

3.3 Компоновщик LD

Выполняю линковку командой ld и получил исполняемый файл. (рис. 3.5)

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

Рис. 3.5: Линковка программы

Еще раз выполняю линковку для объектного файла obj.o и получаю исполняемый файл main.(рис. 3.6)

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

Рис. 3.6: Линковка программы

Запускаю исполняемые файлы.(рис. 3.7)

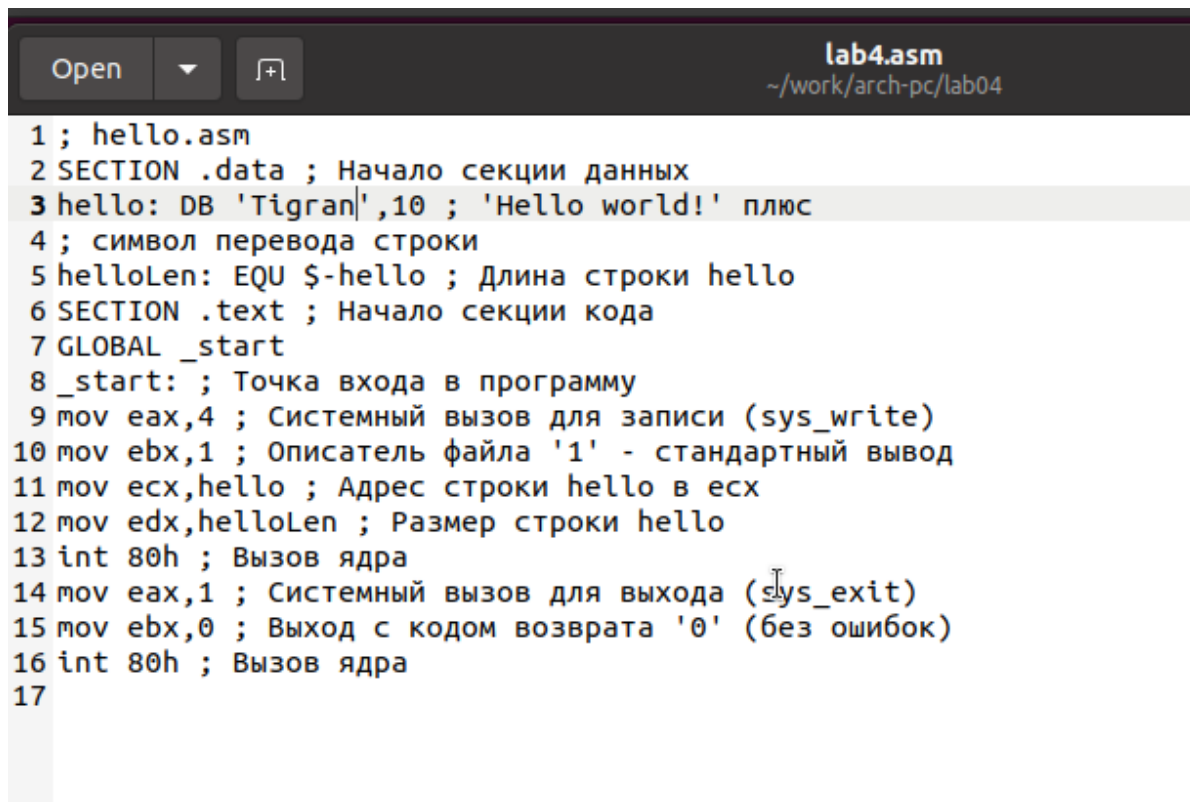
```
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$ ./hello
Hello world!
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$ ./main
Hello world!
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$
```

Рис. 3.7: Запуск программ

3.4 Выполнение заданий для самостоятельной работы.

Копирую программу в новый файл.

Изменяю сообщение Hello world на свое имя (рис. 3.8) и запускаю новую программу. (рис. 3.9)



```
1 ; hello.asm
2 SECTION .data ; Начало секции данных
3 hello: DB 'Tigran',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
17
```

Рис. 3.8: Код программы в файле lab4.asm

```

tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$ cp hello.asm lab4.asm
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$ gedit lab4.asm
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$ nasm -f elf lab4.asm
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$ ls -m elf_i386 lab4.o -o lab4
ls: cannot access 'elf_i386': No such file or directory
ls: cannot access 'lab4': No such file or directory
-rw-rw-r-- 1 tdbabudzyan 640 Oct 18 06:49 lab4.o
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$ nasm -f elf lab4.asm
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$ ld -m elf_i386 lab4.o -o lab4
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$ ls
hello hello.asm hello.o lab4 lab4.asm lab4.o list.lst main obj.o
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$ ./lab4
Tigran
tdbabudzyan@VirtualBox:~/work/arch-pc/lab04$

```

Рис. 3.9: Запуск программы lab4.asm

4 Выводы

При выполнении данной лабораторной работы я освоил процесс компиляции и сборки программ, написанных на ассемблере `nasm`.

Список литературы

1. Архитектура ЭВМ
2. NASM Assembly Language Tutorials. — 2021. — URL: <https://asmtutor.com/>.