

Шаблон отчёта по лабораторной работе

No 5

НВЕ МАНГЕ ХОСЕ ХЕРСОН МИКО, Группа: НКАбд-03-22

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы :	6
2.1	Программа Hello world! :	6
2.2	транслятор NASM :	9
2.3	расширенный синтаксис командной строки NASM :	9
2.4	Компоновщик LD :	10
2.5	запуск исполняемого файла :	10
2.6	Выводы по результатам выполнения заданий:	11
3	Задание для самостоятельной работы :	12
3.1	тестирование написанной программы :	14
3.2	Выводы по результатам выполнения заданий :	16
4	Выводы, согласованные с целью работы :	17

Список иллюстраций

2.1	Ресунок 1	6
2.2	Ресунок 2	7
2.3	Ресунок 3	8
2.4	Ресунок 4	9
2.5	Ресунок 5	9
2.6	Ресунок 6	10
2.7	Ресунок 7	10
2.8	Ресунок 8	10
3.1	Ресунок 9	13
3.2	Ресунок 10	14
3.3	Ресунок 11	15
3.4	Ресунок 12	15
3.5	Ресунок 13	15
3.6	Ресунок 14	16

Список таблиц

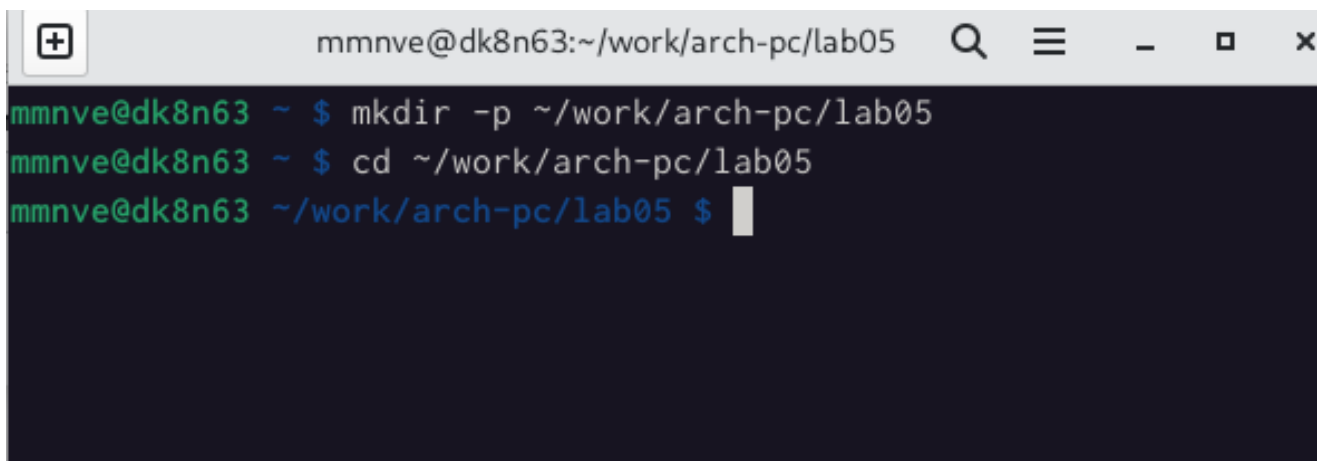
1 Цель работы

В пятой лабораторной работе мы рассмотрим, как освоить процедуру компиляции и сборки программ, написанных на ассемблере `nasm`.

2 Выполнение лабораторной работы :

2.1 Программа Hello world! :

- В этом разделе мы хотели создать программу, которая выводит строку “Hello world!” но на языке ассемблера nasm.
- Вот почему мы начали с рекурсивного создания нового каталога “~/work/arch-pc/lab05” (рис. [-fig. 2.1])

A screenshot of a terminal window with a dark background. The window title bar shows the username 'mmnve' and host 'dk8n63', along with standard window controls. The terminal shows three lines of commands: 'mkdir -p ~/work/arch-pc/lab05', 'cd ~/work/arch-pc/lab05', and a prompt in the directory '~/.work/arch-pc/lab05'.

```
mmnve@dk8n63:~/work/arch-pc/lab05
mmnve@dk8n63 ~ $ mkdir -p ~/work/arch-pc/lab05
mmnve@dk8n63 ~ $ cd ~/work/arch-pc/lab05
mmnve@dk8n63 ~/.work/arch-pc/lab05 $
```

Рис. 2.1: Ресунок 1

- После этого мы создали текстовый файл в формате .asm, затем открываем только что созданный файл с помощью текстового редактора gedit.рис. [-fig. 2.2)

```
mmnve@dk3n66 ~/work/study/2022-2023/Архитектура компьютера/study_2022-2023_arh-p  
c/labs/lab05 $ touch hello.asm  
mmnve@dk3n66 ~/work/study/2022-2023/Архитектура компьютера/study_2022-2023_arh-p  
c/labs/lab05 $ gedit hello.asm  
mmnve@dk3n66 ~/work/study/2022-2023/Архитектура компьютера/study_2022-2023_arh-p  
c/labs/lab05 $
```

Рис. 2.2: Ресунок 2

- После этого мы добавили код сборки, который выводит “Hello world!” в файл hello.asm.рис. [-fig. 2.3)

~/work/study/2024

report(2).md

X

```
1 SECTION .data
2     hello: DB 'Hello world!',10
3
4     helloLen: EQU $-hello
5 SECTION .text
6     GLOBAL _start
7
8 _start:
9 mov eax,4
10 mov ebx,1
11 mov ecx,hello
12 mov edx,helloLen
13 int 80h
14
15 mov eax,1
16 mov ebx,0
17 int 80h
```

Рис. 2.3: Ресунок 3

2.2 ранслятор NASM :

- На этом этапе, используя переводчик NASM, мы смогли скомпилировать или перевести код в объектный код, который создал другой файл с форматом.рис. [-fig. 2.4)

```
mmnve@dk8n63 ~/work/arch-pc/lab05 $ nasm -f elf hello.asm
mmnve@dk8n63 ~/work/arch-pc/lab05 $ ls
hello.asm  hello.o
mmnve@dk8n63 ~/work/arch-pc/lab05 $
```

Рис. 2.4: Ресунок 4

- Используя команду ls, мы проверили работу, проделанную переводчиком, и обнаружили, что объектный файл был создан с тем же именем, что и текстовый файл.

2.3 асширенный синтаксис командной строки NASM :

- Здесь мы запустили полную команду NASM и проверили выходные файлы, которые дала нам.Разница заключалась в том, что с помощью полной команды нам нужно указать имя объектного файла и список файлов, и это то, что получилось после проверки с помощью запятой ls.рис. [-fig. 2.5)

```
hello.asm  hello.o
mmnve@dk8n63 ~/work/arch-pc/lab05 $ nasm -o obj.o -f elf -g -l list.lst hello.asm
mmnve@dk8n63 ~/work/arch-pc/lab05 $ ls
hello.asm  hello.o  list.lst  obj.o
mmnve@dk8n63 ~/work/arch-pc/lab05 $
```

Рис. 2.5: Ресунок 5

2.4 Компоновщик LD :

- На этом шаге и с помощью компоновщика с командой `ld` мы смогли получить исполняемый файл, обработав объектный файл. Затем, используя команду `ls`, мы проверили, что файл был создан. рис. [-fig. 2.6)

```
mmnve@dk8n63 ~/work/arch-pc/lab05 $ ld -m elf_i386 hello.o -o hello
mmnve@dk8n63 ~/work/arch-pc/lab05 $ ls
hello hello.asm hello.o list.lst obj.o
mmnve@dk8n63 ~/work/arch-pc/lab05 $
```

Рис. 2.6: Ресунок 6

- Затем мы проверили, что можем присвоить исполняемому файлу любое имя, а не только то же имя, что и объектному файлу, как показано с помощью команды `ls`.рис. [-fig. 2.7)

```
mmnve@dk8n63 ~/work/arch-pc/lab05 $ ld -m elf_i386 obj.o -o main
mmnve@dk8n63 ~/work/arch-pc/lab05 $ ls
hello hello.asm hello.o list.lst main obj.o
mmnve@dk8n63 ~/work/arch-pc/lab05 $
```

Рис. 2.7: Ресунок 7

- сполняемый файл имеет имя “main”, а для объектного файла - “obj”.

2.5 апуск исполняемого файла :

- На этом шаге все, что мы сделали, это запустили исполняемый файл.рис. [-fig. 2.8)

```
mmnve@dk8n63 ~/work/arch-pc/lab05 $ ./hello
Hello world!
mmnve@dk8n63 ~/work/arch-pc/lab05 $
```

Рис. 2.8: Ресунок 8

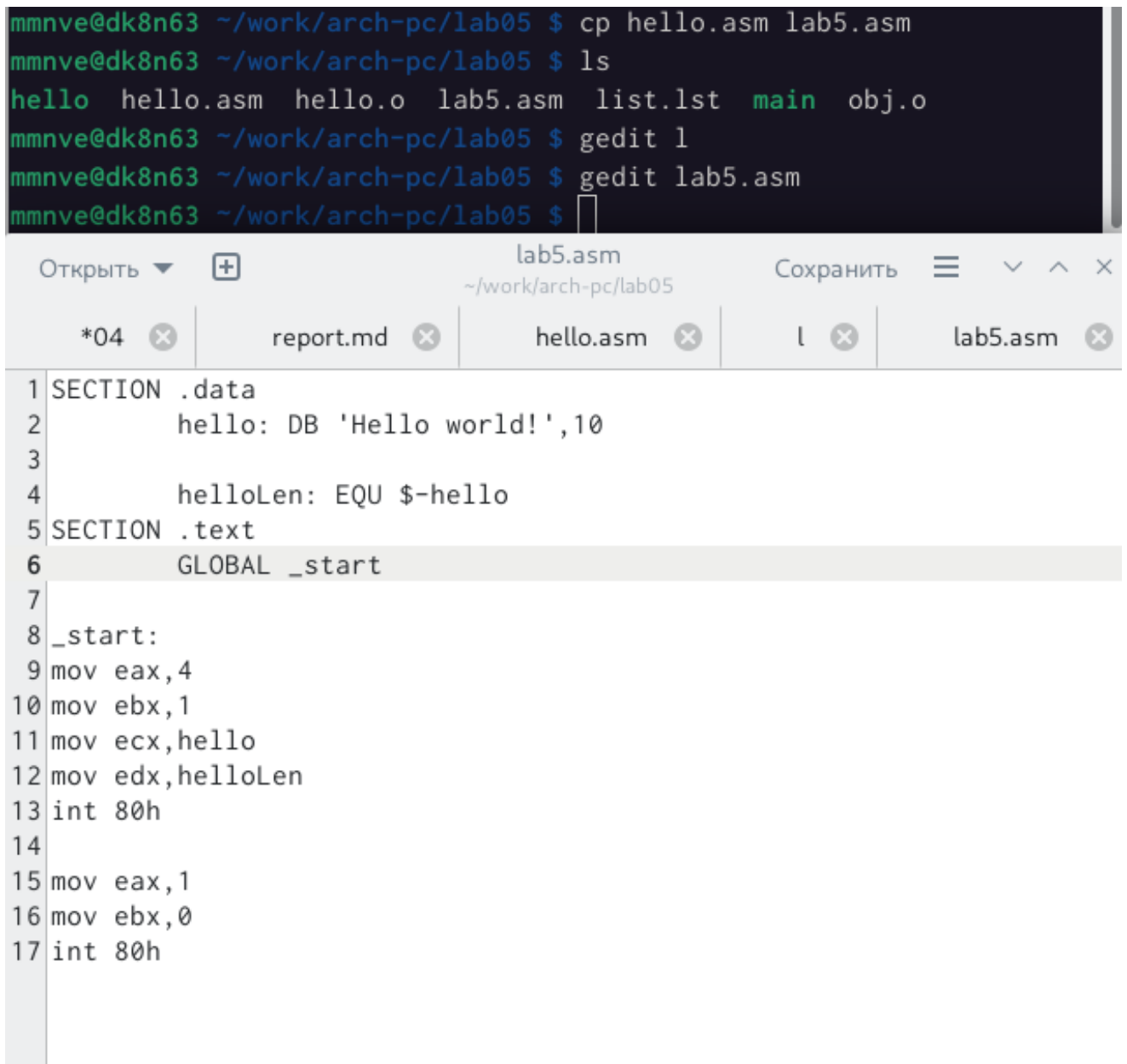
2.6 Выводы по результатам выполнения заданий:

- В этой лабораторной работе мы освоили, как скомпилировать текстовый файл, написанный на языке ассемблера NASM, в объектный файл, затем получить опкод, и все это ради создания программы, которая печатает знаменитое предложение “Hello world!”

3 Задание для самостоятельной работы :

1. В каталоге `~/work/arch-рс/lab05` мы создали копию для файла `hello.asm` и присвоили ему имя `lab05`. рис. [-fig. 3.1)

```
mmnve@dk8n63 ~/work/arch-pc/lab05 $ cp hello.asm lab5.asm
mmnve@dk8n63 ~/work/arch-pc/lab05 $ ls
hello hello.asm hello.o lab5.asm list.lst main obj.o
mmnve@dk8n63 ~/work/arch-pc/lab05 $ gedit l
mmnve@dk8n63 ~/work/arch-pc/lab05 $ gedit lab5.asm
mmnve@dk8n63 ~/work/arch-pc/lab05 $
```



The image shows a terminal window and a gedit editor window. The terminal window displays the following commands and output:

```
mmnve@dk8n63 ~/work/arch-pc/lab05 $ cp hello.asm lab5.asm
mmnve@dk8n63 ~/work/arch-pc/lab05 $ ls
hello hello.asm hello.o lab5.asm list.lst main obj.o
mmnve@dk8n63 ~/work/arch-pc/lab05 $ gedit l
mmnve@dk8n63 ~/work/arch-pc/lab05 $ gedit lab5.asm
mmnve@dk8n63 ~/work/arch-pc/lab05 $
```

The gedit editor window shows the contents of the file lab5.asm. The file contains the following assembly code:

```
1 SECTION .data
2     hello: DB 'Hello world!',10
3
4     helloLen: EQU $-hello
5 SECTION .text
6     GLOBAL _start
7
8 _start:
9 mov eax,4
10 mov ebx,1
11 mov ecx,hello
12 mov edx,helloLen
13 int 80h
14
15 mov eax,1
16 mov ebx,0
17 int 80h
```

Рис. 3.1: Ресунек 9

- Используя текстовый редактор gedit, мы изменили текстовый файл, содержащий ассемблерный код, чтобы программа выводила мое имя и фамилию “Mougari Abderrahim”.рис. [-fig. 3.2)



```
1 SECTION .data
2     hello: DB 'Hello world!',10
3
4     helloLen: EQU $-hello
5 SECTION .text
6     GLOBAL _start
7
8 _start:
9 mov eax,4
10 mov ebx,1
11 mov ecx,hello
12 mov edx,helloLen
13 int 80h
14
15 mov eax,1
16 mov ebx,0
17 int 80h
```

Рис. 3.2: Ресунок 10

3.1 истинг написанной программы :

SECTION .data

nfname: DB 'jose gerson',10

nfnameLen: EQU \$-nfname SECTION .text GLOBAL _start _start:

mov eax,4 mov ebx,1 mov ecx,nfname mov edx,nfnameLen int 80h

mov eax,1 mov ebx,0 int 80h

3. После написания кода е скомпилировал код в объектный файл после чего получил исполняемый файл с помощью компоновщика. рис. [-fig. 3.3)

```
mmnve@dk8n63 ~/work/arch-pc/lab05 $ nasm -f elf lab5.asm
mmnve@dk8n63 ~/work/arch-pc/lab05 $ ls
hello hello.asm hello.o lab5.asm lab5.o list.lst main obj.o
mmnve@dk8n63 ~/work/arch-pc/lab05 $ ld -m elf_i386 lab5.o -o lab5
mmnve@dk8n63 ~/work/arch-pc/lab05 $ ls
hello      hello.o lab5.asm list.lst obj.o
hello.asm lab5      lab5.o   main
mmnve@dk8n63 ~/work/arch-pc/lab05 $
```

Рис. 3.3: Ресунк 11

- Затем мы запустили исполняемый файл. рис. [-fig. 3.4)

```
mmnve@dk8n63 ~/work/arch-pc/lab05 $ ./lab5
Hello world!
mmnve@dk8n63 ~/work/arch-pc/lab05 $
```

Рис. 3.4: Ресунк 12

- Здесь мы скопировали оба hello.Asm и lab5.asm в ваш локальный репозиторий. рис. [-fig. 3.5)

```
mmnve@dk8n63 ~/work/arch-pc/lab05 $ cp hello.asm lab5.asm ~/work/study/2022-2023/Архитектура\ компьютера/study_2022-2023_arh-pc/labs/lab05/
mmnve@dk8n63 ~/work/arch-pc/lab05 $ cd ~/work/study/2022-2023/Архитектура\ компьютера/study_2022-2023_arh-pc/labs/lab05/
mmnve@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/study_2022-2023_arh-pc/labs/lab05 $ ls
hello.asm lab5.asm presentation report
mmnve@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/study_2022-2023_arh-pc/labs/lab05 $
```

Рис. 3.5: Ресунк 13

- Наконец, мы загрузили все файлы в удаленный репозиторий. рис. [-fig. 3.6)

```

mmnve@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/study_2022-2023_arh-pc/labs/lab05 $ git add .
mmnve@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/study_2022-2023_arh-pc/labs/lab05 $ git commit -am 'feat(main): add files lab-5'
[master 247ef5c] feat(main): add files lab-5
5 files changed, 216 insertions(+), 3 deletions(-)
create mode 100644 labs/lab05/hello.asm
create mode 100644 labs/lab05/lab5.asm
create mode 100644 labs/lab05/report/04
create mode 100644 labs/lab05/report/Без имени 1
mmnve@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/study_2022-2023_arh-pc/labs/lab05 $ git push
Перечисление объектов: 17, готово.
Подсчет объектов: 100% (17/17), готово.
При сжатии изменений используется до 6 потоков
Сжатие объектов: 100% (10/10), готово.
Запись объектов: 100% (10/10), 1.08 КиБ | 1.08 МБ/с, готово.
Всего 10 (изменений 5), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (5/5), completed with 5 local objects.
To github.com:josegersonmikonve/study_2022-2023_arh-pc.git
 e2a6789..247ef5c master -> master
mmnve@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/study_2022-2023_arh-pc/labs/lab05 $

```

Рис. 3.6: Ресунок 14

3.2 Выводы по результатам выполнения заданий :

- В этих упражнениях мы применили навыки, полученные в ходе лабораторной работы, в ходе которой получили более глубокое представление об именах регистров и о том, как выделить для них память.

4 ыводы, согласованные с целью работы

:

- В шестой лабораторной работе мы можем получить практические навыки по созданию компиляции и обработке программы с использованием языка ассемблера Nasm.