

Отчёт по лабораторной работе№4

Создание и процесс обработки программ на языке ассемблера NASM

Марцев Аркадий Алексеевич

Содержание

1	Цель работы	1
2	Задание	1
3	Выполнение лабораторной работы.....	1
4	Выводы	5

1 Цель работы

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

2 Задание

Здесь приводится описание задания в соответствии с рекомендациями методического пособия и выданным вариантом.

3 Выполнение лабораторной работы

1. Создайте каталог для работы с программами на языке ассемблера NASM:

```
ааmarcev@ааmarcev-Modern-14-C12M:~$ mkdir -p ~/work/arch-pc/lab04
```

Рис. 1

Создание каталога

2. Перейдём в созданный каталог:

```
ааmarcev@ааmarcev-Modern-14-C12M:~$ cd ~/work/arch-pc/lab04
```

Рис. 2

Переход в каталог

3. Создадим текстовый файл с именем hello.asm:

```
ааmarcev@ааmarcev-Modern-14-C12M:~/work/arch-pc/lab04$ touch hello.asm
ааmarcev@ааmarcev-Modern-14-C12M:~/work/arch-pc/lab04$ ls
hello.asm
```

Рис. 3

Создание текстового файла

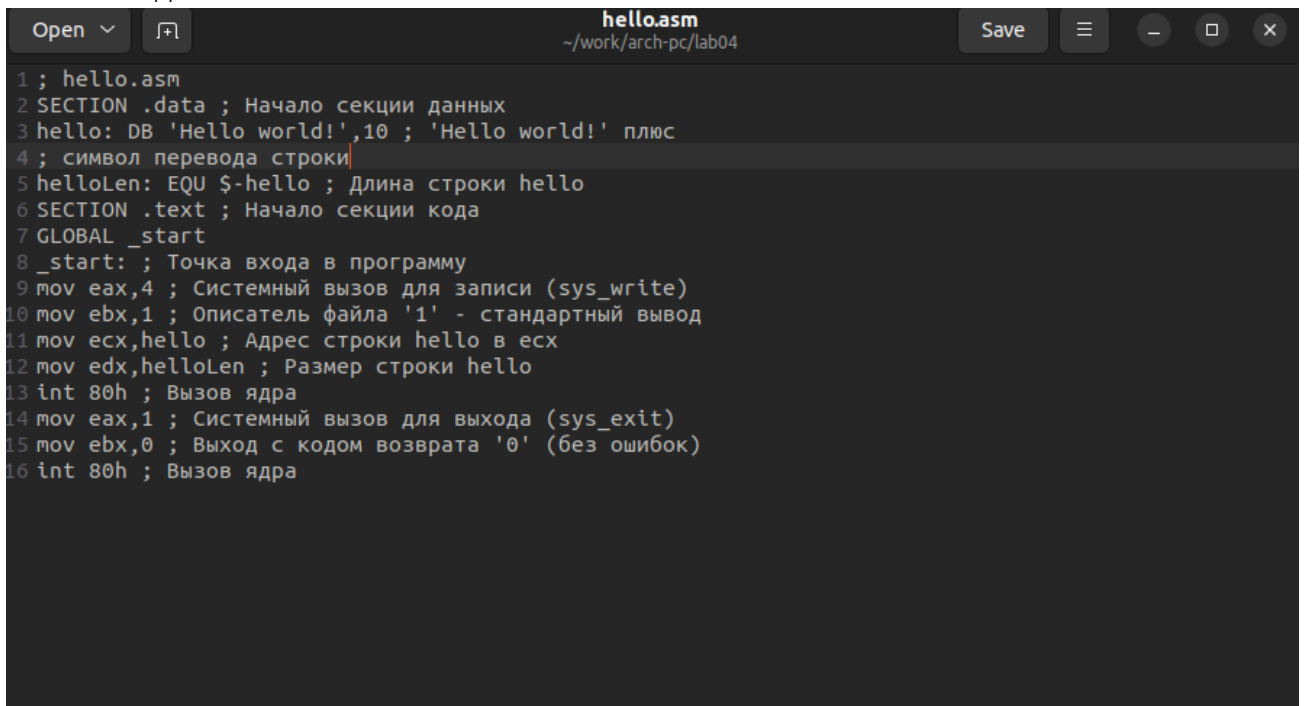
4. Откроем этот файл с помощью текстового редактора

```
ааmarcev@ааmarcev-Modern-14-C12M:~/work/arch-pc/lab04$ gedit hello.asm
```

Рис. 4

Открытие файла

5. Введём в него текст:



```
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 есх
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. 5

Ввод текста

6. Скомпилируем данный текст

```
ааmarcev@ааmarcev-Modern-14-C12M:~/work/arch-pc/lab04$ nasm -f elf hello.asm
```

Рис.6

Компиляция текста

7. Проверим, что объектный файл был создан:

```
ааmarcev@ааmarcev-Modern-14-C12M:~/work/arch-pc/lab04$ ls  
hello.asm hello.o
```

Рис. 7

проверка, что объектный файл был создан

8. Скомпилируем исходный файл hello.asm в obj.o и создадим файл листинга list.lst

```
ааmarcev@ааmarcev-Modern-14-C12M:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.lst hello.a  
sm
```

Рис. 8

Создание файлов

9. Проверим, что файлы были созданы.

```
ааmarcev@ааmarcev-Modern-14-C12M:~/work/arch-pc/lab04$ ls  
hello.asm hello.o list.lst obj.o
```

Рис. 9

Проверка, что файлы были созданы.

10. Передадим объектный файл на обработку компоновщику.

```
ааmarcev@ааmarcev-Modern-14-C12M:~/work/arch-pc/lab04$ ld -m elf_i386 hello.o -o hello
```

Рис. 10

Передача файла на компоновку

11. Проверим, что исполняемый файл hello был создан.

```
ааmarcev@ааmarcev-Modern-14-C12M:~/work/arch-pc/lab04$ ls  
hello hello.asm hello.o list.lst obj.o
```

Рис. 11

Проверка, что исполняемый файл hello был создан

12. Зададим имя создаваемого исполняемого файла.

```
ааmarcev@ааmarcev-Modern-14-C12M:~/work/arch-pc/lab04$ ld -m elf-i386 obj.o -o main
```

Рис. 12

Зададим имя создаваемого исполняемого файла

13. Запустим на выполнение созданный исполняемый файл, находящийся в текущем каталоге.

```
ааmarcev@ааmarcev-Modern-14-C12M:~/work/arch-pc/lab04$ ./hello  
Hello world!
```

Рис.13

Запуск на выполнение созданный исполняемый файл

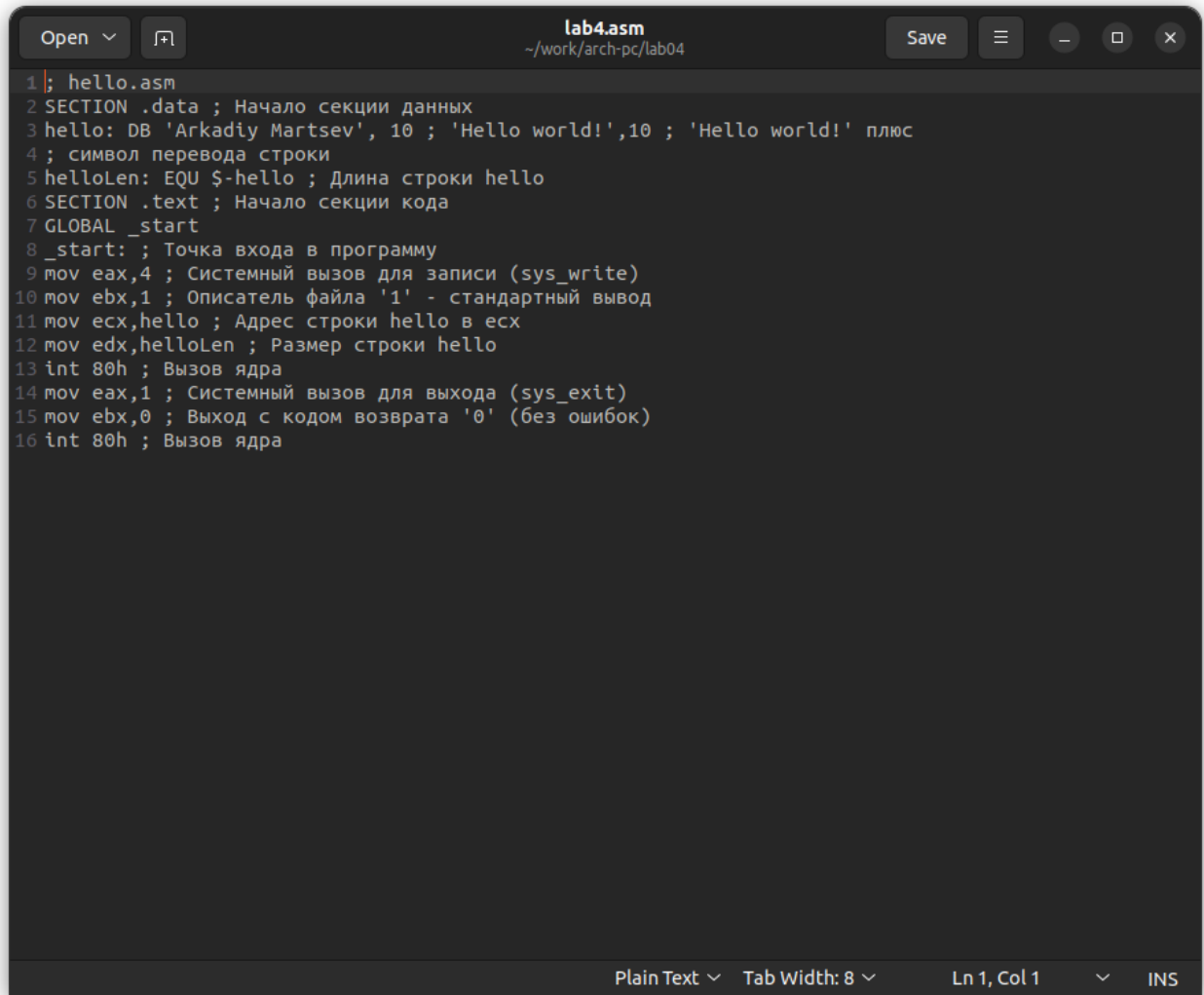
14. Создадим копию файла hello.asm с именем lab4.asm

```
aamarcev@aamarcev-Modern-14-C12M:~/work/arch-pc/lab04$ cp hello.asm lab4.asm
```

Рис. 14

Создание копии файла с именем lab4.asm

15. Внесём изменения в текст программы в файле lab5.asm



```
lab4.asm
~/work/arch-pc/lab04

1 |; hello.asm
2 |SECTION .data ; Начало секции данных
3 |hello: DB 'Arkadiy Martsev', 10 ; '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. 15

Внесение изменения в текст программы

16. Оттранслируем полученный текст программы lab5.asm в объектный файл. Выполним компоновку объектного файла и запустим получившийся исполняемый файл.

```
aamarcev@aamarcev-Modern-14-C12M:~/work/arch-pc/lab04$ gedit hello.asm
aamarcev@aamarcev-Modern-14-C12M:~/work/arch-pc/lab04$ cp hello.asm lab4.asm
aamarcev@aamarcev-Modern-14-C12M:~/work/arch-pc/lab04$ nasm -f elf lab4.asm
aamarcev@aamarcev-Modern-14-C12M:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.lst lab4.asm
aamarcev@aamarcev-Modern-14-C12M:~/work/arch-pc/lab04$ nasm -o Arkadiy.o -f elf -g -l list.lst lab4.asm
aamarcev@aamarcev-Modern-14-C12M:~/work/arch-pc/lab04$ ld -m elf_i386 Arkadiy.o -o Arkadiy
aamarcev@aamarcev-Modern-14-C12M:~/work/arch-pc/lab04$ ./Arkadiy
Arkadiy Martsev
```

Рис. 16

Оттранслирование, компоновка, запуск

17. Скопировала файлы hello.asm и lab4.asm в локальный репозиторий в каталог ~/work/study/2023-2024/“Архитектура компьютера”/arch-pc/labs/lab04/ с помощью утилиты cp и проверил наличие файлов с помощью утилиты ls

```
aamarcev@aamarcev-Modern-14-C12M:~$ cd ~/work/arch-pc/lab04
aamarcev@aamarcev-Modern-14-C12M:~/work/arch-pc/lab04$ cp hello.asm ~/work/study/23-24/'Computer architecture'/arch-pc/labs/lab04
aamarcev@aamarcev-Modern-14-C12M:~/work/arch-pc/lab04$ cp lab04.asm ~/work/study/23-24/'Computer architecture'/arch-pc/labs/lab04
cp: cannot stat 'lab04.asm': No such file or directory
aamarcev@aamarcev-Modern-14-C12M:~/work/arch-pc/lab04$ cp lab4.asm ~/work/study/23-24/'Computer architecture'/arch-pc/labs/lab04
aamarcev@aamarcev-Modern-14-C12M:~/work/arch-pc/lab04$ cd ~/work/study/23-24/'Computer architecture'/arch-pc/labs/lab04
aamarcev@aamarcev-Modern-14-C12M:~/work/study/23-24/Computer architecture/arch-pc/labs/lab04$ ls
hello.asm lab4.asm presentation report
```

Рис. 17

Копирование файлов в локальный репозиторий

18. Загружаю файлы на Github

```
aamarcev@aamarcev-Modern-14-C12M:~/work/study/23-24/Computer architecture/arch-pc/labs/lab04/report$ ls
bib hello.asm image lab4.asm Makefile pandoc report.md
aamarcev@aamarcev-Modern-14-C12M:~/work/study/23-24/Computer architecture/arch-pc/labs/lab04/report$ git add .
aamarcev@aamarcev-Modern-14-C12M:~/work/study/23-24/Computer architecture/arch-pc/labs/lab04/report$ git commit -am 'feat(main): make course structure'
[master 0a911d3] feat(main): make course structure
 2 files changed, 32 insertions(+)
   create mode 100644 labs/lab04/report/hello.asm
   create mode 100644 labs/lab04/report/lab4.asm
aamarcev@aamarcev-Modern-14-C12M:~/work/study/23-24/Computer architecture/arch-pc/labs/lab04/report$ git push
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 12 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 1.04 KiB | 1.04 MiB/s, done.
Total 6 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To github.com:aamarcev/pcarch_NBI01-23.git
   cda40d7..0a911d3 master -> master
```

Рис. 18

Загрузка файлов на гитхаб

4 Выводы

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