Шаблон отчёта по лабораторной работе

5

Сильвен Макс Грегор Филс , НКАбд-03-22

Содержание

# 1 Цель работы

* В пятой лабораторной работе мы рассмотрим, как освоить процедуру компи- ляции и сборки программ, написанных на ассемблере nasm.

# 2 Выполнение лабораторной работы :

* 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. 1)

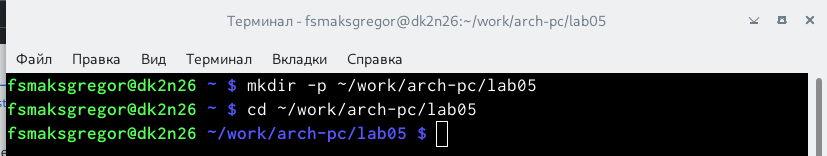


Рис. 1: Ресукнек 1

* После этого мы создали текстовый файл в формате .asm, затем открываем только что созданный файл с помощью текстового редактора gedit.(рис. 2)

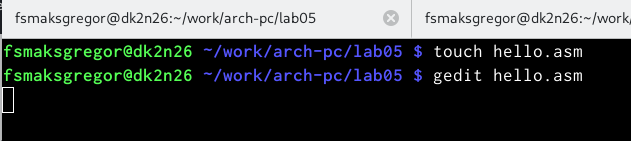


Рис. 2: Ресукнек 2

* После этого мы добавили код сборки, который выводит “Hello world!” в файл hello.asm. (рис. 3)

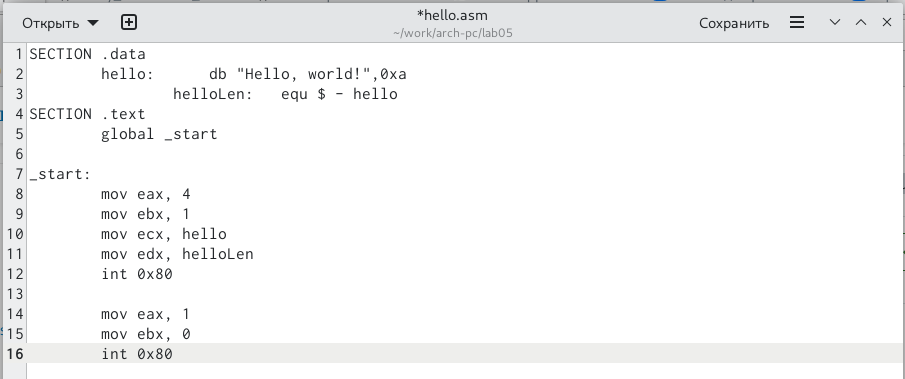


Рис. 3: Ресукнек 3

## 2.1 Транслятор NASM :

* На этом этапе, используя переводчик NASM, мы смогли скомпилировать или перевести код в объектный код, который создал другой файл с форматом **.o.**.(рис. 4)

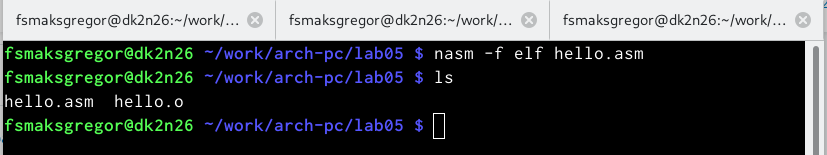


Рис. 4: Ресукнек 4

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

### 2.1.1 Расширенный 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 ко- манды нам нужно указать имя объектного файла и список файлов, и это то, что получилось после проверки с помощью запятой **ls**.(рис. 5)

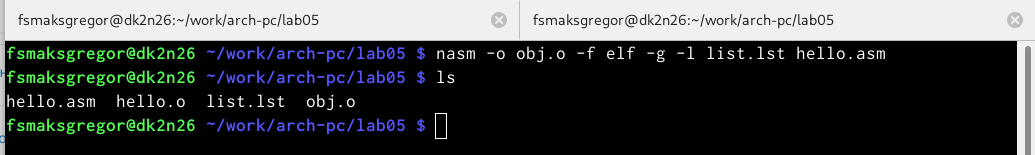


Рис. 5: Ресукнек 5

#### 2.1.1.1 Компоновщик LD :

* На этом шаге и с помощью компоновщика с командой ld мы смогли по- лучить исполняемый файл, обработав объектный файл. Затем,используя команду ls, мы проверили, что файл был создан.(рис. 6)

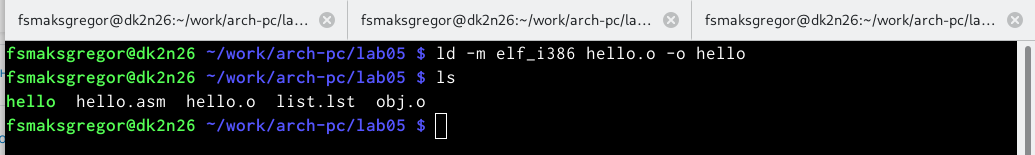


Рис. 6: Ресукнек 6

* 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. 7)

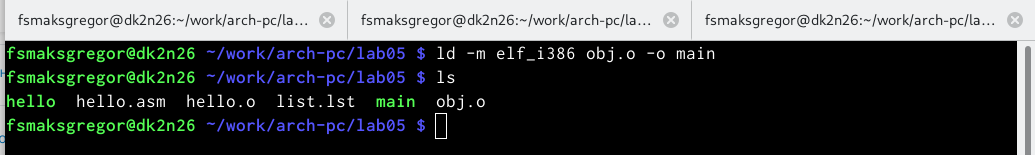


Рис. 7: Ресукнек 7

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

##### 2.1.1.1.1 Запуск исполняемого файла :

* На этом шаге все, что мы сделали, это запустили исполняемый файл. (рис. 8)

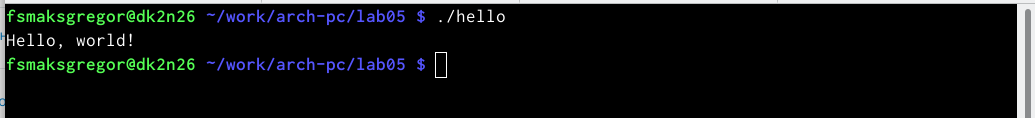


Рис. 8: Ресукнек 8

###### 2.1.1.1.1.1 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 оправдание, и все это ради создания программы, которая печатает знаменитое предложение “**Hello world!**”

# 3 Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы.

Например, в табл. 1 приведено краткое описание стандартных каталогов Unix.

Таблица 1: Описание некоторых каталогов файловой системы GNU Linux

| Имя каталога | Описание каталога |
| --- | --- |
| / | Корневая директория, содержащая всю файловую |
| /bin | Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям |
| /etc | Общесистемные конфигурационные файлы и файлы конфигурации установленных программ |
| /home | Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя |
| /media | Точки монтирования для сменных носителей |
| /root | Домашняя директория пользователя root |
| /tmp | Временные файлы |
| /usr | Вторичная иерархия для данных пользователя |

Более подробно об Unix см. в [1–6].

# 4 Задание для самостоятельной работы :

* В каталоге ~/work/arch-pc/lab05 мы создали копию для файла hello.asm и присвоили ему имя lab05. (рис. 9)

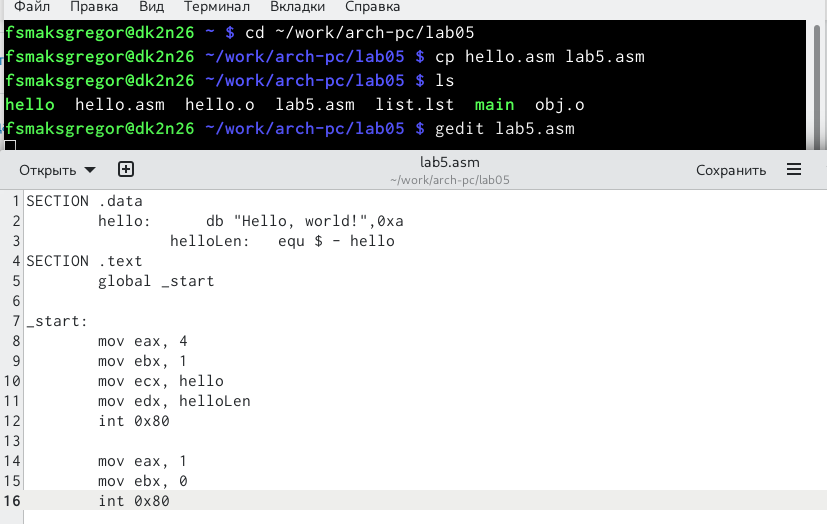


Рис. 9: Ресукнек 9

* Используя текстовый редактор gedit, мы изменили текстовый файл, содержащий ассемблерный код, чтобы программа выводила мое имя и фамилию “Max Sylvain”.

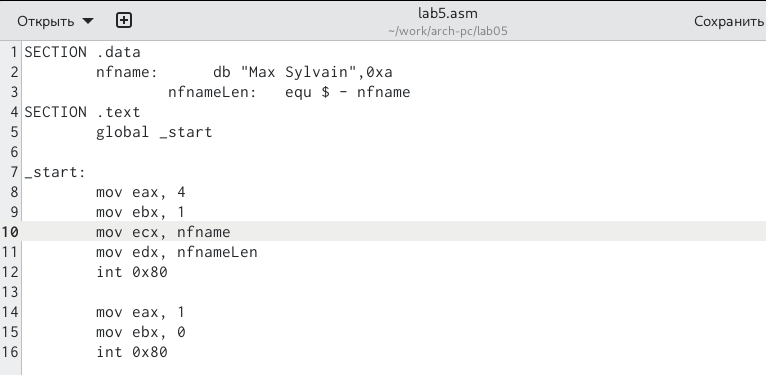


Рис. 10: Ресукнек 10

* После написания кода e скомпилировал код в объектный файл после чего получил исполняемый файл с помощью компоновщика. (рис. 11)

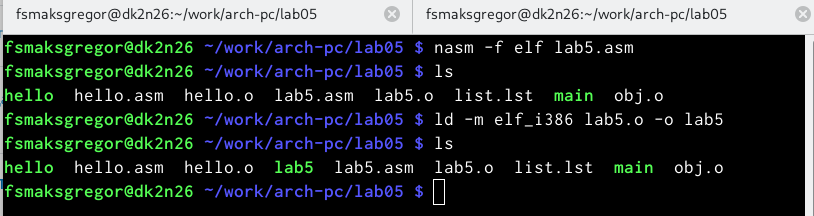


Рис. 11: Ресукнек 11

* Затем мы запустили исполняемый файл.(рис. 12)

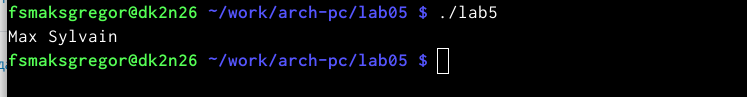


Рис. 12: Ресукнек 12

* Здесь мы скопировали оба hello.Asm и lab5.asm в ваш локальный репози- торий.(рис. 13)

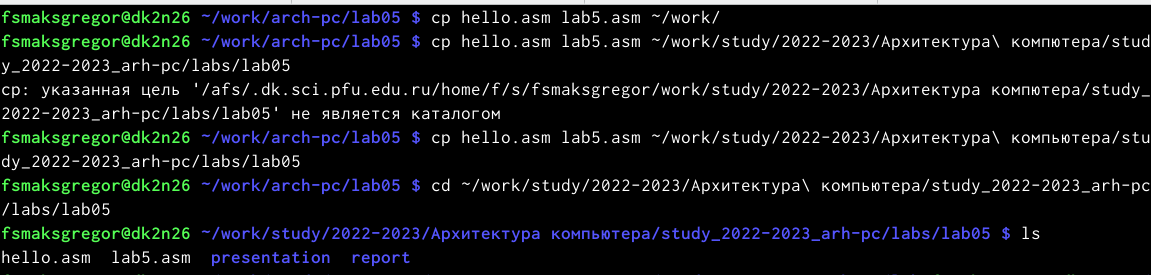


Рис. 13: Ресукнек 13

## 4.1 Выводы по результатам выполнения заданий :

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

# 5 Выводы

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

# Список литературы

1. GNU Bash Manual [Электронный ресурс]. Free Software Foundation, 2016. URL: <https://www.gnu.org/software/bash/manual/>.

2. Newham C. [Learning the bash Shell: Unix Shell Programming](http://www.amazon.com/Learning-bash-Shell-Programming-Nutshell/dp/0596009658). O’Reilly Media, 2005. 354 с.

3. Zarrelli G. Mastering Bash. Packt Publishing, 2017. 502 с.

4. Robbins A. [Bash Pocket Reference](https://www.ncbi.nlm.nih.gov/pubmed/25246403). O’Reilly Media, 2016. 156 с.

5. Таненбаум Э. Архитектура компьютера. 6-е изд. СПб.: Питер, 2013. 874 с.

6. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.