Шаблон отчёта по лабораторной работе №5

Мугари Абдеррахим , НКАбд-03-22

Содержание

# 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. 1)

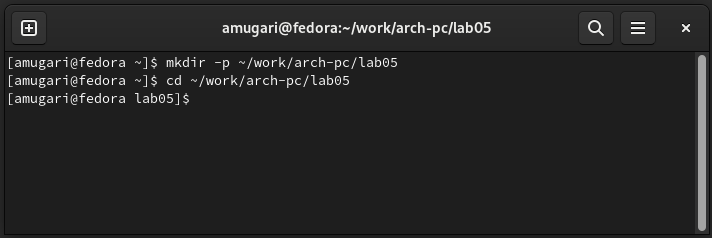


Рис. 1: Ресунок 1

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

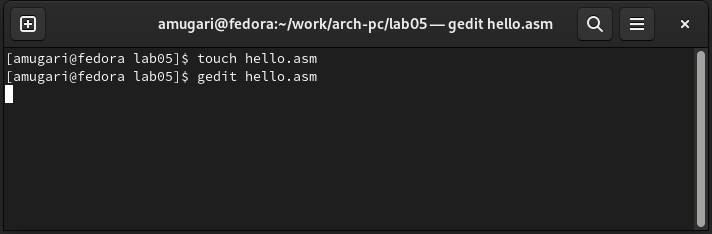


Рис. 2: Ресунок 2

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

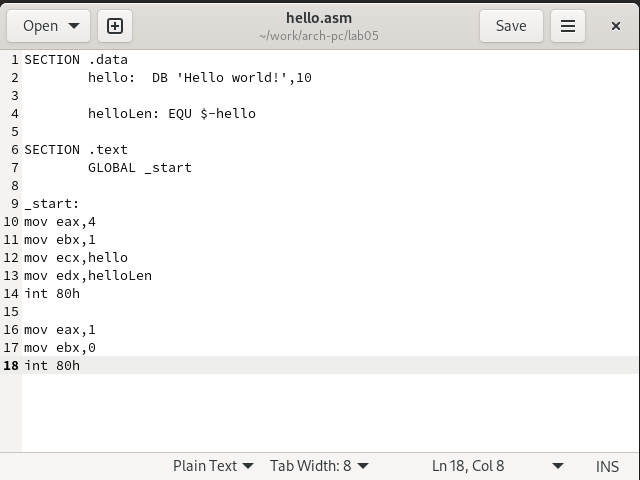


Рис. 3: Ресунок 3

## 2.2 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)

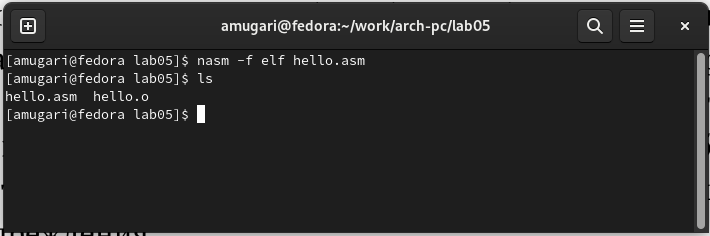


Рис. 4: Ресунок 4

* 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 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 команды нам нужно указать имя объектного файла и список файлов, и это то, что получилось после проверки с помощью запятой **ls**. (рис. 5)

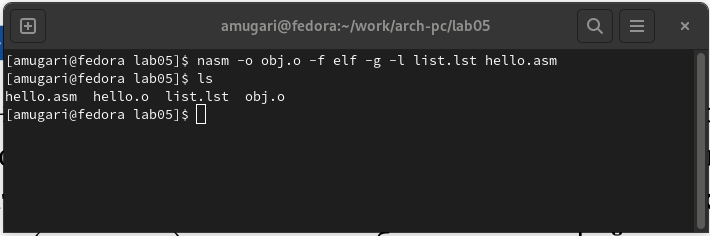


Рис. 5: Ресунок 5

## 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. 6)

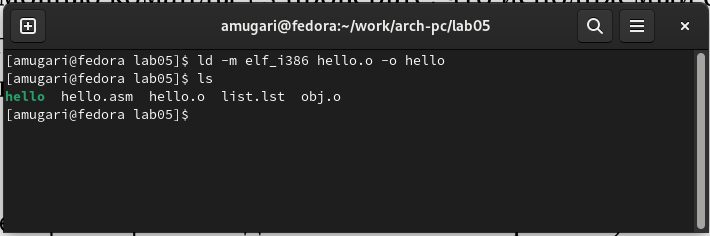


Рис. 6: Ресунок 6

* 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. 7)

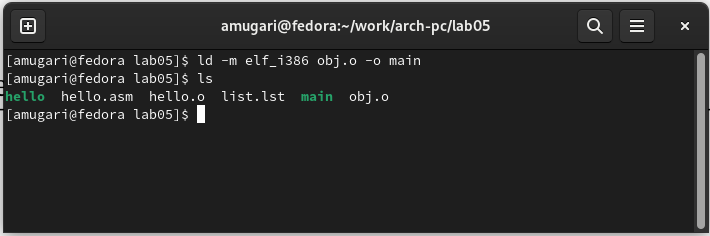


Рис. 7: Ресунок 7

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

## 2.5 Запуск исполняемого файла :

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

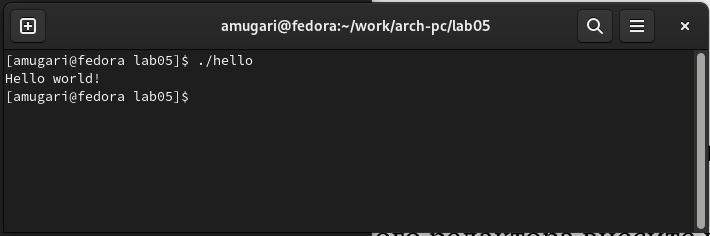


Рис. 8: Ресунок 8

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

# 3 Задание для самостоятельной работы :

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

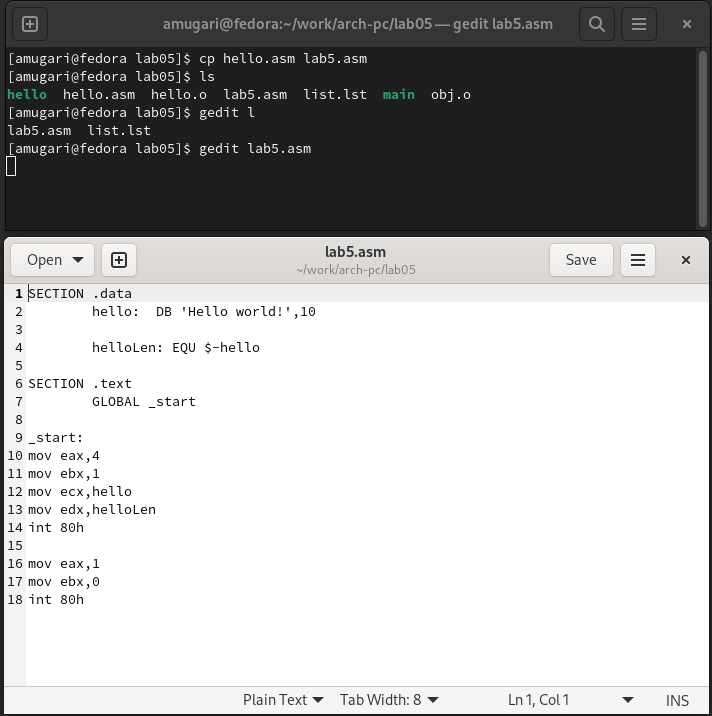


Рис. 9: Ресунок 9

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

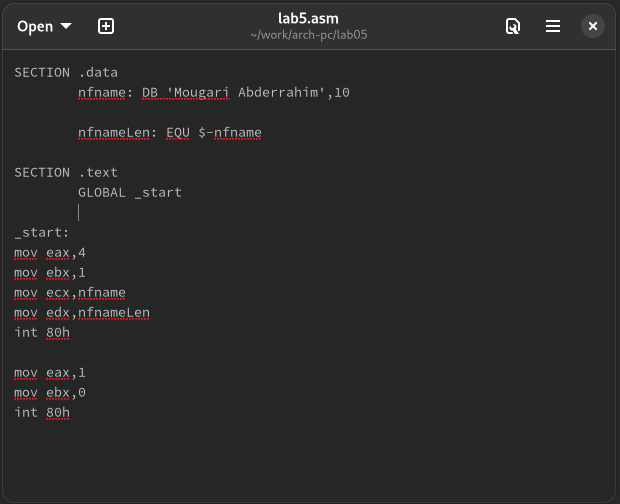


Рис. 10: Ресунок 10

### 3.0.1 листинг написанной программы :

SECTION .data

nfname: DB ‘Mougari Abderrahim’,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

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

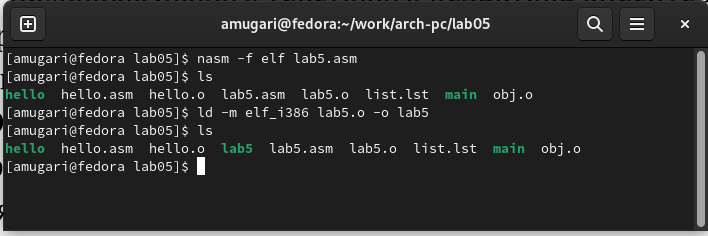


Рис. 11: Ресунок 11

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

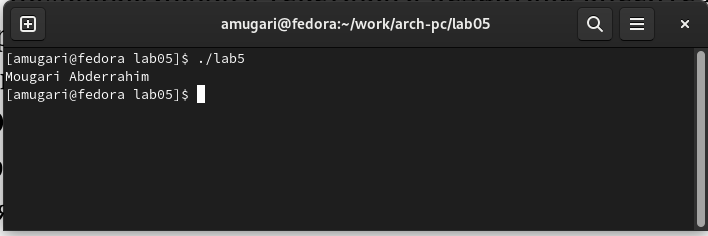


Рис. 12: Ресунок 12

1. Здесь мы скопировали оба **hello.Asm** и **lab5.asm** в ваш локальный репозиторий. (рис. 13)

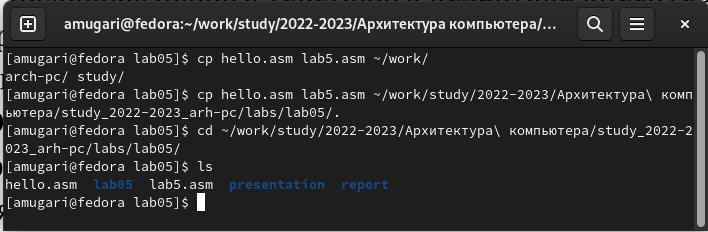


Рис. 13: Ресунок 13

* Наконец, мы загрузили все файлы в удаленный репозиторий. (рис. 14)

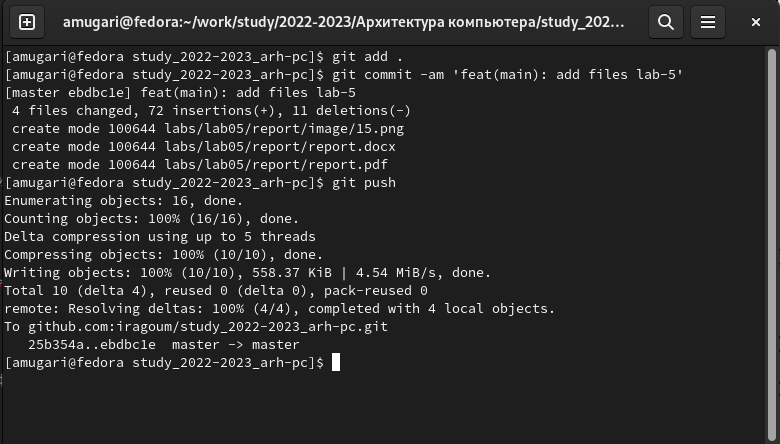


Рис. 14: Ресунок 14

## 3.1 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 Выводы, согласованные с целью работы :

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