

Отчет по лабораторной работе №5

Дисциплина: архитектура компьютера

Астраханцева Анастасия Александровна

Содержание

1	Цель работы	4
2	Задание	5
3	Выполнение лабораторной работы	6
4	Выводы	10

Список иллюстраций

3.1	Создание папки lab05 и файла hello.asm	6
3.2	Запись текста программы	7
3.3	Компиляция текста программы	7
3.4	Копирование hello.asm с именем lab5.asm	8
3.5	Изменение текста программы	8
3.6	Вывод текста программы	9
3.7	Перенос файлов	9
3.8	Загрузка на GitHub	9

1 Цель работы

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

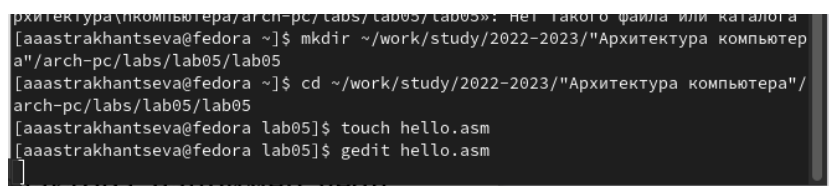
2 Задание

Знакомство с машинно-ориентированным языком низкого уровня - ассемблером,
написание первой программы “Hello world!”

3 Выполнение лабораторной работы

Программа Hello world!

Для начала создам папку lab05 в ~/work/study/2022-2023/“Архитектура компьютера”/arch-pc/labs/lab05. После этого перейду в эту папку. Далее создам текстовый файл hello.asm и открою его с помощью gedit(рис. 3.1):



```
Архитектура(компьютера)/arch-pc/labs/lab05/lab05»: нет такого файла или каталога
[aaastrakhantseva@fedora ~]$ mkdir ~/work/study/2022-2023/"Архитектура компьютер
a"/arch-pc/labs/lab05/lab05
[aaastrakhantseva@fedora ~]$ cd ~/work/study/2022-2023/"Архитектура компьютера"/
arch-pc/labs/lab05/lab05
[aaastrakhantseva@fedora lab05]$ touch hello.asm
[aaastrakhantseva@fedora lab05]$ gedit hello.asm
```

Рис. 3.1: Создание папки lab05 и файла hello.asm

После открытия ввожу в это файл текст программы (рис. 3.2):

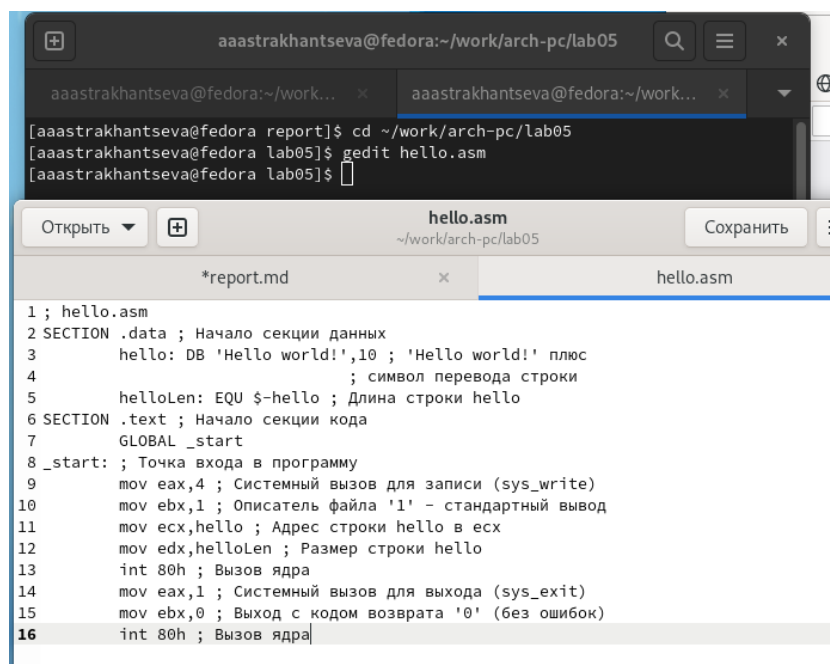


Рис. 3.2: Запись текста программы

Транслятор NASM

NASM превращает текст программы в объектный код. Для компиляции программы «Hello World» необходимо написать такую команду (рис. 3.3):

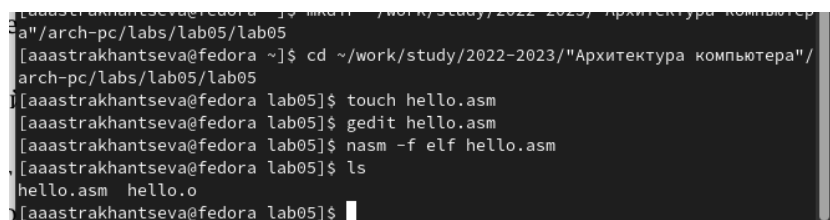


Рис. 3.3: Компиляция текста программы

Транслятор преобразует текст программы из файла hello.asm в объектный код, который запишется в файл hello.o. NASM не запускают без параметров, ключ -f указывает транслятору, что требуется создать бинарные файлы в формате ELF. Формат elf64 позволяет создавать исполняемый код, работающий под 64-битными версиями Linux. NASM всегда создаёт выходные файлы в текущем каталоге.

Компоновщик LD

Передаю объектный файл компоновщику (рис. ??). После этого запускаю на выполнение созданный исполняемый файл:

Обработка программы компоновщиком

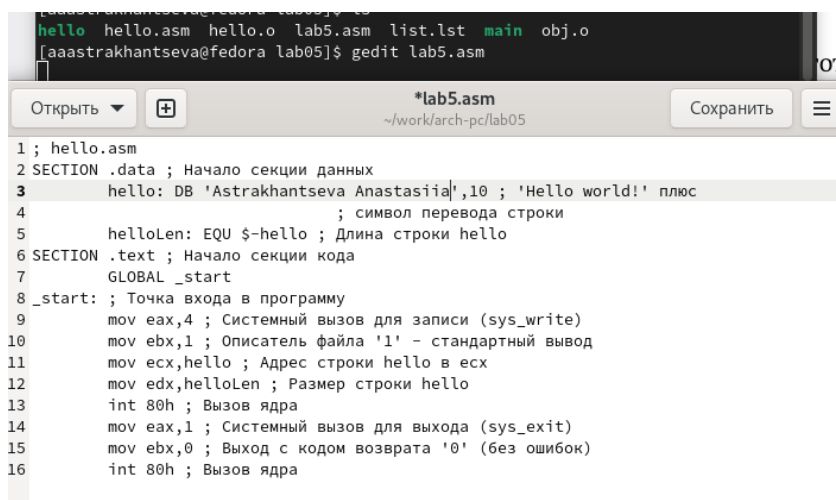
Задания для самостоятельной работы

Файл hello.asm я перенесла из ~/work/study/2022-2023/“Архитектура компьютера” ~/arch-pc/labs/lab05 в ~/work/arch-pc/lab05. После этого я создала копию этого файла с именем lab5.asm (рис. 3.4):

```
[aaastrakhantseva@fedora ~]$ cd work/arch-pc/lab05
[aaastrakhantseva@fedora lab05]$ ls
hello hello.asm hello.o list.lst main obj.o
[aaastrakhantseva@fedora lab05]$ cp hello.asm lab5.asm
[aaastrakhantseva@fedora lab05]$ ls
hello hello.asm hello.o lab5.asm list.lst main obj.o
[aaastrakhantseva@fedora lab05]$
```

Рис. 3.4: Копирование hello.asm с именем lab5.asm

С помощью gedit вношу изменения в написанную программу, чтобы она выводила мое имя и фамилию (рис. 3.5):



```
hello hello.asm hello.o lab5.asm list.lst main obj.o
[aaastrakhantseva@fedora lab05]$ gedit lab5.asm

*lab5.asm
~/work/arch-pc/lab05

1 ; hello.asm
2 SECTION .data ; Начало секции данных
3     hello: DB 'Astrakhantseva Anastasiia',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. 3.5: Изменение текста программы

Далее нужно оттранслировать полученный текст программы lab5.asm в объектный файл. После этого запускаю получившийся файл (рис. 3.6):


```

[aaastrakhantseva@fedora lab05]$ nasm -f elf lab5.asm
[aaastrakhantseva@fedora lab05]$ ls
hello hello.asm hello.o lab5.asm lab5.o list.lst main obj.o
[aaastrakhantseva@fedora lab05]$ nasm -o obj.o -f elf -g -l list.lst lab5.asm
[aaastrakhantseva@fedora lab05]$ ls
hello hello.asm hello.o lab5.asm lab5.o list.lst main obj.o
[aaastrakhantseva@fedora lab05]$ ld -m elf_i386 lab5.o -o lab5
[aaastrakhantseva@fedora lab05]$ ld -m elf_i386 obj.o -o main
[aaastrakhantseva@fedora lab05]$ ./lab5
Astrakhantseva Anastasiia
[aaastrakhantseva@fedora lab05]$ w

```

Рис. 3.6: Вывод текста программы

Далее копирую получившиеся файлы из папки ~/arch-pc/labs/lab05 в ~/work/study/2022-2023/“Архитектура компьютера”/arch-pc/labs/lab05 с помощью команды cp (рис. 3.7):

```

ла или каталога
[aaastrakhantseva@fedora lab05]$ cd
[aaastrakhantseva@fedora ~]$ cp work/arch-pc/lab05/hello.asm work/study/2022-2023/
3/"Архитектура компьютера"/arch-pc/labs/lab05
[aaastrakhantseva@fedora ~]$ cp work/arch-pc/lab05/lab5.asm work/study/2022-2023/
/"Архитектура компьютера"/arch-pc/labs/lab05
[aaastrakhantseva@fedora ~]$ cd work/study/2022-2023/"Архитектура компьютера"/ar
ch-pc/labs/lab05
[aaastrakhantseva@fedora lab05]$ ls
hello.asm lab05 lab5.asm presentation report
[aaastrakhantseva@fedora lab05]$

```

Рис. 3.7: Перенос файлов

После этого зашгружаю файлы на GitHub (рис. 3.8):

```

[aaastrakhantseva@fedora ~]$ cd ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc
[aaastrakhantseva@fedora arch-pc]$ git add .
[aaastrakhantseva@fedora arch-pc]$ git commit -am 'feat(main): add files lab-5'
[master 0f53f97] feat(main): add files lab-5
12 files changed, 48 insertions(+)
create mode 100644 labs/lab04/report.zip
create mode 100644 labs/lab04/report/image.zip
create mode 100644 labs/lab05/hello.asm
create mode 100644 labs/lab05/lab05
create mode 100644 labs/lab05/lab5.asm
create mode 100644 labs/lab05/report/image/hello.o.png
create mode 100644 labs/lab05/report/image/main & hello.png
create mode 100644 labs/lab05/report/image/peremeshenie.png
create mode 100644 labs/lab05/report/image/saml.png
create mode 100644 labs/lab05/report/image/saml.png
create mode 100644 labs/lab05/report/image/zapysk.png
create mode 100644 labs/lab05/report/image/создание папки хеллоу асм.png
[aaastrakhantseva@fedora arch-pc]$ git push
Перечисление объектов: 26, готово.
Подсчет объектов: 100% (26/26), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (19/19), готово.
Запись объектов: 100% (19/19), 5.88 Миб | 265.00 КиБ/с, готово.
Всего 19 (изменений 6), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (6/6), completed with 4 local objects.
To github.com:anastasiia7205/study_2022-2023_arh-pc.git
ce9e4d3..0f53f97 master -> master
[aaastrakhantseva@fedora arch-pc]$

```

Рис. 3.8: Загрузка на GitHub

4 Выводы

Я познакомилась с языком ассемблер, освоила процедуры компиляции и сборки программ, написанных на ассемблере NASM.