

Отчёт по лабораторной работе №5

Создание и процесс обработки программ на языке ассемблера NASM

Дарья Сергеевна Кочина

Содержание

1	Цель работы	4
2	Задание	5
3	Теоретическое введение	6
4	Выполнение лабораторной работы	7
5	Выводы	11

Список иллюстраций

4.1	Создание каталога lab05 и переход к нему	7
4.2	Файл hello.asm	7
4.3	Текст в файле hello.asm	7
4.4	Компиляция текста	8
4.5	Компиляция файла hello.asm в obj.o	8
4.6	Обработка компоновщиком	8
4.7	Запуск файла	8
4.8	Копия файла hello.asm	8
4.9	Изменение текста в файле lab05.asm	9
4.10	Изменение файла	9
4.11	Запуск файла lab05	9
4.12	Копирование файлов	9
4.13	Загрузка на GitHub	10

1 Цель работы

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

2 Задание

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

3 Теоретическое введение

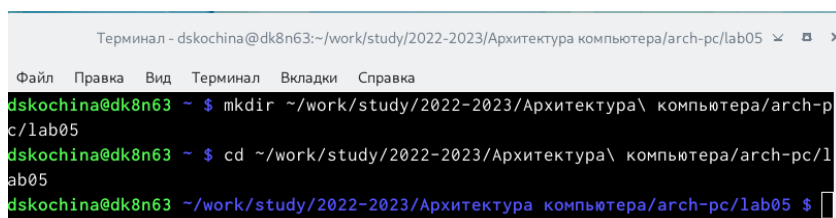
Язык ассемблера — машинно-ориентированный язык низкого уровня. Можно считать, что он больше любых других языков приближен к архитектуре ЭВМ и её аппаратным возможностям, что позволяет получить к ним более полный доступ, нежели в языках высокого уровня, таких как C/C++, Perl, Python и пр. Таким образом, язык ассемблера — это язык, с помощью которого понятным для человека образом пишутся команды для процессора.

Преобразование или трансляция команд с языка ассемблера в исполняемый машинный код осуществляется специальной программой транслятором — **Ассемблер**.

NASM — это открытый проект ассемблера, версии которого доступны под различные операционные системы и который позволяет получать объектные файлы для этих систем. В NASM используется Intel-синтаксис и поддерживаются инструкции x86-64.

4 Выполнение лабораторной работы

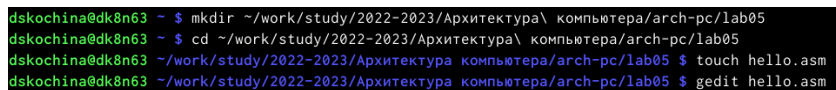
1. Я создала каталог lab05 для работы с программами на языке ассемблера NASM, затем перешла в него. (рис. 4.1)



```
Терминал - dskochina@dk8n63:~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05
Файл  Правка  Вид  Терминал  Вкладки  Справка
dskochina@dk8n63 ~ $ mkdir ~/work/study/2022-2023/Архитектура\ компьютера/arch-pc/lab05
dskochina@dk8n63 ~ $ cd ~/work/study/2022-2023/Архитектура\ компьютера/arch-pc/lab05
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $
```

Рис. 4.1: Создание каталога lab05 и переход к нему

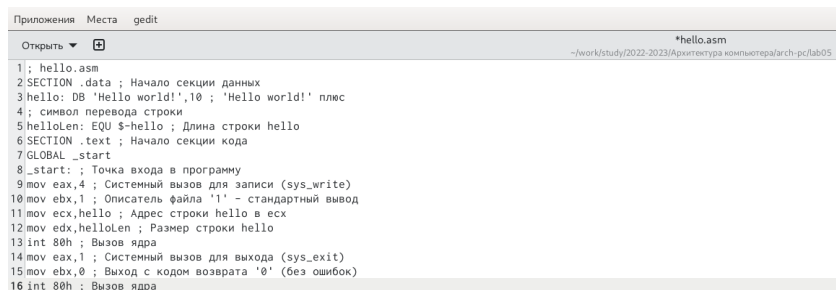
2. Я создала файл hello.asm и открыла его с помощью текстового редактора gedit. (рис. 4.2)



```
dskochina@dk8n63 ~ $ mkdir ~/work/study/2022-2023/Архитектура\ компьютера/arch-pc/lab05
dskochina@dk8n63 ~ $ cd ~/work/study/2022-2023/Архитектура\ компьютера/arch-pc/lab05
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ touch hello.asm
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ gedit hello.asm
```

Рис. 4.2: Файл hello.asm

3. Я ввела в файл нужный текст. (рис. 4.3)



```
Приложения  Места  gedit
Открыть ▼
*hello.asm
~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05
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 ; Описание файла '1' - стандартный вывод
11: mov ecx,hello ; Адрес строки hello в еск
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. 4.3: Текст в файле hello.asm

4. Я скомпилировала текст с помощью транслятора NASM. (рис. 4.4)

```
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ nasm -f elf hello.asm
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ls
hello.asm  hello.o
```

Рис. 4.4: Компиляция текста

5. Я скомпилировала файл hello.asm в obj.o. (рис. 4.5)

```
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ nasm -o obj.o -f elf -g -l list.lst hello.asm
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ls
hello.asm  hello.o  list.lst  obj.o
```

Рис. 4.5: Компиляция файла hello.asm в obj.o

6. Обработка компоновщиком файла hello.o и obj.o. (рис. 4.6)

```
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ld -m elf_i386 hello.o -o hello
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ls
hello  hello.asm  hello.o  list.lst  obj.o
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ld -m elf_i386 obj.o -o main
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ls
hello  hello.asm  hello.o  list.lst  main  obj.o
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $
```

Рис. 4.6: Обработка компоновщиком

7. Я запустила файл hello. (рис. 4.7)

```
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ./hello
Hello world!
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $
```

Рис. 4.7: Запуск файла

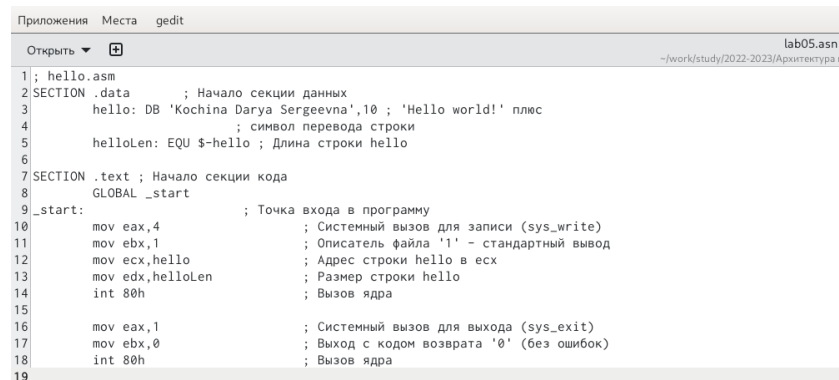
Выполнение самостоятельной работы

1. С помощью команды `cp` я сделала копию файла `hello.asm` с именем `lab05.asm`. (рис. 4.8)

```
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ cp hello.asm lab5.asm
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ls
hello  hello.asm  hello.o  lab5.asm  list.lst  main  obj.o
```

Рис. 4.8: Копия файла hello.asm

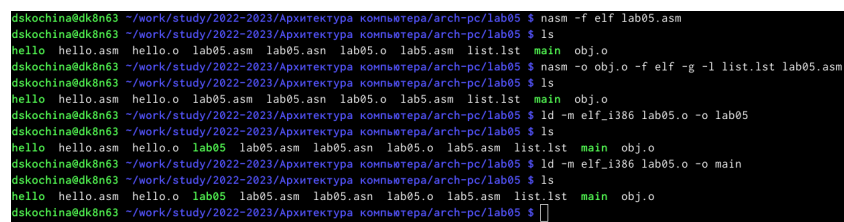
2. С помощью текстового редактора gedit я внесла изменения в файл lab05.asm, чтобы он выводил моё имя и фамилию. (рис. 4.9)



```
1; hello.asm
2SECTION .data ; Начало секции данных
3    hello: DB 'Kochina Darya Sergeevna',10 ; 'Hello world!' плмс
4           ; символ перевода строки
5    helloLen: EQU $-hello ; Длина строки hello
6
7SECTION .text ; Начало секции кода
8    GLOBAL _start
9_start: ; Точка входа в программу
10     mov eax,4 ; Системный вызов для записи (sys_write)
11     mov ebx,1 ; Описатель файла '1' - стандартный вывод
12     mov ecx,hello ; Адрес строки hello в есх
13     mov edx,helloLen ; Размер строки hello
14     int 80h ; Вызов ядра
15
16     mov eax,1 ; Системный вызов для выхода (sys_exit)
17     mov ebx,0 ; Выход с кодом возврата '0' (без ошибок)
18     int 80h ; Вызов ядра
19
```

Рис. 4.9: Изменение текста в файле lab05.asm

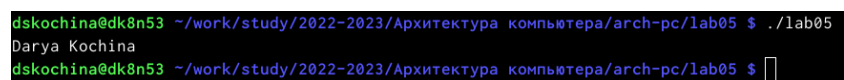
3. Я оттранслировала полученный текст и выполнила компоновку файла. (рис. 4.10)



```
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ nasm -f elf lab05.asm
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ls
hello hello.asm hello.o lab05.asm lab05.asn lab05.o lab5.asm list.lst main obj.o
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ nasm -o obj.o -f elf -g -l list.lst lab05.asm
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ls
hello hello.asm hello.o lab05.asm lab05.asn lab05.o lab5.asm list.lst main obj.o
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ld -m elf_i386 lab05.o -o lab05
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ls
hello hello.asm hello.o lab05.asm lab05.asn lab05.o lab5.asm list.lst main obj.o lab05
dskochina@dk8n63 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $
```

Рис. 4.10: Изменение файла

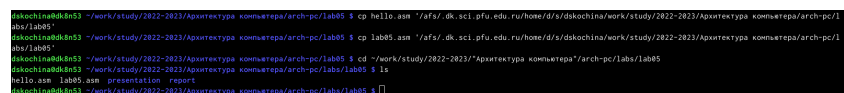
4. Я запустила файл lab05. (рис. 4.11)



```
dskochina@dk8n53 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ./lab05
Darya Kochina
dskochina@dk8n53 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $
```

Рис. 4.11: Запуск файла lab05

5. Я скопировала файлы в локальный репозиторий. (рис. 4.12)



```
dskochina@dk8n53 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ cp hello.asm lab05.asm lab05.asn lab05.o lab5.asm list.lst main obj.o lab05
dskochina@dk8n53 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $ ls
hello hello.asm hello.o lab05.asm lab05.asn lab05.o lab5.asm list.lst main obj.o lab05
dskochina@dk8n53 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/lab05 $
```

Рис. 4.12: Копирование файлов

6. Я загрузила файлы на GitHub. (рис. 4.13)

```
dskochina@dk8n53 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ git add .
dskochina@dk8n53 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ git commit 'lab05.asm hello.asm'
error: pathspec 'lab05.asm hello.asm' did not match any file(s) known to git
dskochina@dk8n53 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $ git commit -am 'lab05.asm hello.asm'
[master 36fa4d6] lab05.asm hello.asm
13 files changed, 119 insertions(+)
create mode 100755 lab05/hello
create mode 100644 lab05/hello.asm
create mode 100644 lab05/hello.o
create mode 100755 lab05/lab05
create mode 100644 lab05/lab05.asm
create mode 100644 lab05/lab05.asn
create mode 100644 lab05/lab05.o
create mode 100644 lab05/lab5.asm
create mode 100644 lab05/list.lst
create mode 100755 lab05/main
create mode 100644 lab05/obj.o
create mode 100644 labs/lab05/hello.asm
create mode 100644 labs/lab05/lab05.asm
dskochina@dk8n53 ~/work/study/2022-2023/Архитектура компьютера/arch-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% (14/14), готово.
Запись объектов: 100% (14/14), 3.10 КиБ | 3.10 МБ/с, готово.
Всего 14 (изменений 7), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (7/7), completed with 2 local objects.
To github.com:dskochina/study_2022-2023_arch-pc.git
f9cb7dc..36fa4d6 master -> master
dskochina@dk8n53 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab05 $
```

Рис. 4.13: Загрузка на GitHub

5 Выводы

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