

Лабораторная работа №5

Архитектура компьютера

Голованова Мария Константиновна

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
4.1	Программа Hello world!	8
4.2	Транслятор NASM	9
4.3	Расширенный синтаксис командной строки NASM	9
4.4	Компоновщик LD	10
4.5	Запуск исполняемого файла	10
5	Задание для самостоятельной работы	12
6	Выводы	15

Список иллюстраций

4.1	Создание пустого каталога, переход в него	8
4.2	Создание текстового файла и его открытие	8
4.3	Ввод текста	9
4.4	Создание объектного файла и проверка действия	9
4.5	Создание файлов obj.o и list.lst и проверка действия	10
4.6	Создание исполняемого файла hello и проверка действия	10
4.7	Создание исполняемого файла main и проверка действия	10
4.8	Запуск созданного исполняемого файла на выполнение	11
5.1	Создание копии файла hello.asm с именем lab5.asm	12
5.2	Создание копии файла hello.asm с именем lab5.asm	12
5.3	Создание объектного файла и проверка действия	13
5.4	Создание файлов obj5.o и list5.lst и проверка действия	13
5.5	Создание исполняемого файла hello и проверка действия	13
5.6	Создание исполняемого файла main5, проверка действия и запуск созданного исполняемого файла на выполнение	13
5.7	Копирование файлов hello.asm и lab5.asm в локальный репозиторий и загрузка файлов на Github	14

Список таблиц

1 Цель работы

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

2 Задание

Создать программу, которая выводит приветственное сообщение Hello world! на экран. Создать программу, выводящую на экран строку с милой фамилией и именем.

3 Теоретическое введение

Язык ассемблера (assembly language, сокращённо asm) — машинно-ориентированный язык низкого уровня, с помощью которого понятным для человека образом пишутся команды для процессора. 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 (C/, C++, Perl, Python и др). Преобразование или трансляция команд с языка ассемблера в исполняемый машинный код осуществляется специальной программой-транслятором — Ассемблером. Ассемблер NASM (Netwide Assembler) — 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

4.1 Программа Hello world!

Я создала каталог для работы с программами на языке ассемблера NASM и перешла в созданный каталог.(рис. 4.1).

```
[mkgolovanova@fedora ~]$  
[mkgolovanova@fedora ~]$ mkdir -p ~/work/arch-pc/lab05  
[mkgolovanova@fedora ~]$ cd ~/work/arch-pc/lab05  
[mkgolovanova@fedora lab05]$
```

Рис. 4.1: Создание пустого каталога, переход в него

Я создала текстовый файл с именем hello.asm и открыла этот файл с помощью текстового редактора gedit (рис. 4.2).

```
mkgolovanova@fedora ~]$ mkdir -p ~/work/arch-pc/lab05  
mkgolovanova@fedora ~]$ cd ~/work/arch-pc/lab05  
mkgolovanova@fedora lab05]$ touch hello.asm  
mkgolovanova@fedora lab05]$ gedit hello.asm
```

Рис. 4.2: Создание текстового файла и его открытие

Я ввела в файл текст (рис. 4.3).

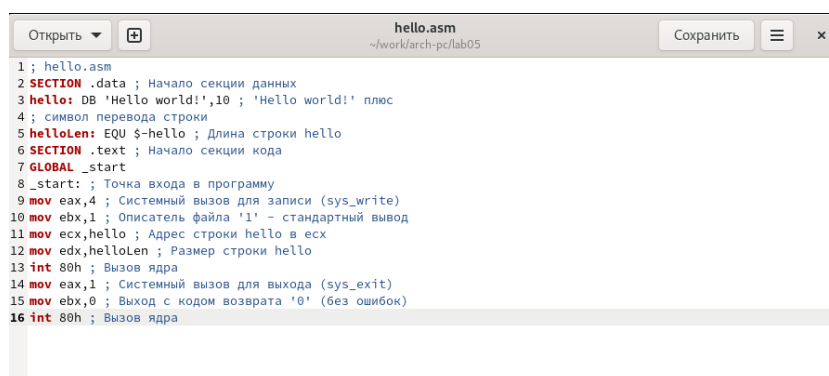


Рис. 4.3: Ввод текста

4.2 Транслятор NASM

Я провела компиляцию приведённого выше текста программы «Hello World» и с помощью команды `ls` проверила, что объектный файл был создан (рис. 4.4).

```

[mkgolovanova@fedora lab05]$ nasm -f elf hello.asm
[mkgolovanova@fedora lab05]$ ls
hello.asm hello.o
[mkgolovanova@fedora lab05]$

```

Рис. 4.4: Создание объектного файла и проверка действия

Транслятор преобразовал текст программы из файла `hello.asm` в объектный код, который записался в данный объектный файл. Объектный файл имеет имя `hello.o`.

4.3 Расширенный синтаксис командной строки NASM

Я скомпилировала исходный файл `hello.asm` в `obj.o` и создала файл листинга `list.lst`, выполнив команду `nasm -o obj.o -f elf -g -l list.lst hello.asm`, и с помощью команды `ls` проверила, что файлы были созданы (рис. 4.5).

```
[mkgolovanova@fedora lab05]$ nasm -o obj.o -f elf -g -l list.lst hello.asm
[mkgolovanova@fedora lab05]$ ls
hello.asm hello.o list.lst obj.o
[mkgolovanova@fedora lab05]$
```

Рис. 4.5: Создание файлов obj.o и list.lst и проверка действия

4.4 Компоновщик LD

Я передала объектный файл на обработку компоновщику и с помощью команды ls проверила, что исполняемый файл hello был создан (рис. 4.6).

```
[mkgolovanova@fedora lab05]$ ld -m elf_i386 hello.o -o hello
[mkgolovanova@fedora lab05]$ ls
hello hello.asm hello.o list.lst obj.o
[mkgolovanova@fedora lab05]$
```

Рис. 4.6: Создание исполняемого файла hello и проверка действия

Я выполнила команду ld -m elf_i386 obj.o -o main и с помощью команды ls проверила, что исполняемый файл был создан (рис. 4.7). Исполняемый файл был собран из файла obj.o и имеет имя main.

```
hello hello.asm hello.o list.lst obj.o
[mkgolovanova@fedora lab05]$ ld -m elf_i386 obj.o -o main
[mkgolovanova@fedora lab05]$ ls
hello hello.asm hello.o list.lst main obj.o
[mkgolovanova@fedora lab05]$
```

Рис. 4.7: Создание исполняемого файла main и проверка действия

4.5 Запуск исполняемого файла

Я запустила на выполнение созданный исполняемый файл, находящийся в текущем каталоге, набрав в командной строке (рис. 4.8): ./hello Программа вывела на экран сообщение Hello world!

```
[mkgolovanova@fedora lab05]$ ./hello  
Hello world!  
[mkgolovanova@fedora lab05]$
```

Рис. 4.8: Запуск созданного исполняемого файла на выполнение

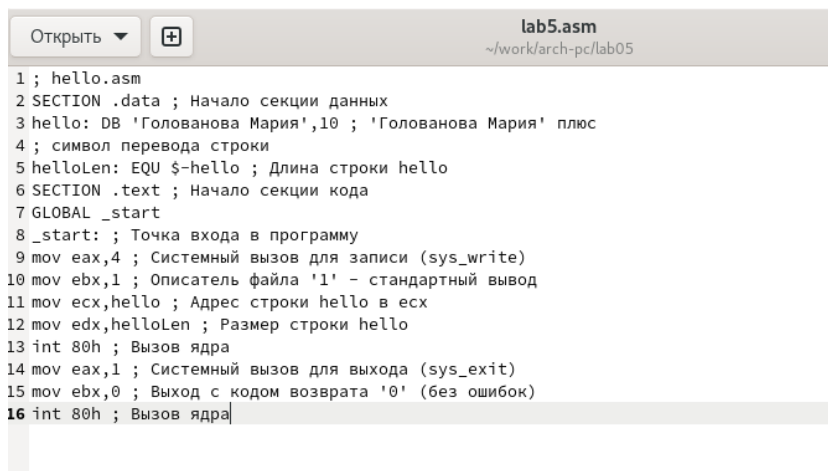
5 Задание для самостоятельной работы

1. Я создала в каталоге ~/work/arch-pc/lab05 с помощью команды `cp` копию файла `hello.asm` с именем `lab5.asm` (рис. 5.1).

```
[mkgolovanova@fedora lab05]$ cp hello.asm lab5.asm
[mkgolovanova@fedora lab05]$ ls
hello  hello.asm  hello.o  lab5.asm  list.lst  main  obj.o
[mkgolovanova@fedora lab05]$
```

Рис. 5.1: Создание копии файла `hello.asm` с именем `lab5.asm`

2. С помощью текстового редактора `gedit` я внесла изменения в текст программы в файле `lab5.asm` так, чтобы вместо `Hello world!` на экран выводилась строка с моими фамилией и именем (рис. 5.2).



```
Открыть ▾ + lab5.asm
~/work/arch-pc/lab05
1 ; hello.asm
2 SECTION .data ; Начало секции данных
3 hello: DB 'Голованова Мария',10 ; 'Голованова Мария' 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. 5.2: Создание копии файла `hello.asm` с именем `lab5.asm`

3. Я оттранслировала полученный текст программы lab5.asm в объектный файл, выполнила компоновку объектного файла и запустила получившийся исполняемый файл (рис. 5.3, рис. 5.4, рис. 5.5, рис. 5.6).

```
[mkgolovanova@fedora lab05]$ nasm -f elf lab5.asm
[mkgolovanova@fedora lab05]$ ls
hello hello.asm hello.o lab5.asm lab5.o list.lst main obj.o
```

Рис. 5.3: Создание объектного файла и проверка действия

```
[mkgolovanova@fedora lab05]$ nasm -o obj5.o -f elf -g -l list5.lst lab5.asm
[mkgolovanova@fedora lab05]$ ls
hello hello.o lab5.o list.lst obj5.o
hello.asm lab5.asm list5.lst main obj.o
[mkgolovanova@fedora lab05]$
```

Рис. 5.4: Создание файлов obj5.o и list5.lst и проверка действия

```
hello.asm lab5.asm list5.lst main obj.o
[mkgolovanova@fedora lab05]$ ld -m elf_i386 lab5.o -o lab5
[mkgolovanova@fedora lab05]$ ls
hello hello.o lab5.asm list5.lst main obj.o
hello.asm lab5 lab5.o list.lst obj5.o
```

Рис. 5.5: Создание исполняемого файла hello и проверка действия

```
[mkgolovanova@fedora lab05]$ ld -m elf_i386 obj5.o -o main5
[mkgolovanova@fedora lab05]$ ls
hello hello.o lab5.asm list5.lst main obj5.o
hello.asm lab5 lab5.o list.lst main5 obj.o
[mkgolovanova@fedora lab05]$ ./lab5
Голованова Мария
[mkgolovanova@fedora lab05]$
```

Рис. 5.6: Создание исполняемого файла main5, проверка действия и запуск созданного исполняемого файла на выполнение

4. Я скопировала файлы hello.asm и lab5.asm в мой локальный репозиторий в каталог ~/work/study/2022-2023/“Архитектура компьютера”/arch-rc/labs/lab05/ и загрузила файлы на Github (рис. 5.7).

```

[mkgolovanova@fedora lab05]$ cp ~/work/arch-pc/lab05/hello.asm ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc/labs/lab05/hello.asm
[mkgolovanova@fedora lab05]$ cp ~/work/arch-pc/lab05/lab5.asm ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc/labs/lab05/lab5.asm
[mkgolovanova@fedora lab05]$ cd ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc/labs/lab05/
[mkgolovanova@fedora lab05]$ ls
hello.asm  lab5.asm  presentation  report
[mkgolovanova@fedora lab05]$ cd ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc/
[mkgolovanova@fedora arch-pc]$ git add .
[mkgolovanova@fedora arch-pc]$ git commit -am 'files in lab05'
[master a4a2d38] files in lab05
17 files changed, 83 insertions(+), 37 deletions(-)
create mode 100644 labs/lab05/hello.asm
create mode 100644 labs/lab05/lab5.asm
create mode 100644 labs/lab05/report/image/Снимок экрана 2022-11-10 105421.png
create mode 100644 labs/lab05/report/image/Снимок экрана 2022-11-10 110726.png
create mode 100644 labs/lab05/report/image/Снимок экрана 2022-11-10 110801.png
create mode 100644 labs/lab05/report/image/Снимок экрана 2022-11-10 111701.png
create mode 100644 labs/lab05/report/image/Снимок экрана 2022-11-10 111902.png
create mode 100644 labs/lab05/report/image/Снимок экрана 2022-11-10 112540.png
create mode 100644 labs/lab05/report/image/Снимок экрана 2022-11-10 112805.png
create mode 100644 labs/lab05/report/image/Снимок экрана 2022-11-10 113048.png
create mode 100644 labs/lab05/report/image/Снимок экрана 2022-11-10 215411.png
create mode 100644 labs/lab05/report/image/Снимок экрана 2022-11-10 220745.png
create mode 100644 labs/lab05/report/image/Снимок экрана 2022-11-10 222457.png
create mode 100644 labs/lab05/report/image/Снимок экрана 2022-11-10 222706.png
create mode 100644 labs/lab05/report/image/Снимок экрана 2022-11-10 223137.png
create mode 100644 labs/lab05/report/image/Снимок экрана 2022-11-10 225508.png
[mkgolovanova@fedora arch-pc]$ git push
Перечисление объектов: 29, готово.
Подсчет объектов: 100% (29/29), готово.
Сжатие объектов: 100% (23/23), готово.
Запись объектов: 100% (23/23), 217.08 КиБ | 561.00 КиБ/с, готово.
Всего 23 (изменений 5), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (5/5), completed with 4 local objects.
To github.com:MKGolovanova/study_2022-2023_arh-pc.git
ce55e6a..a4a2d38 master -> master
[mkgolovanova@fedora arch-pc]$

```

Рис. 5.7: Копирование файлов hello.asm и lab5.asm в локальный репозиторий и загрузка файлов на Github

6 Выводы

Я освоила процедуры компиляции и сборки программ, написанных на ассемблере NASM.