

Лабораторная работа №5

Создание и процесс обработки программ на языке ассемблера NASM

Вершинина Ангелина Алексеевна

Содержание

1	Цель работы	5
2	Теоретическое введение	6
3	Выполнение лабораторной работы	7
3.1	Программа Hello world!	7
3.2	Транслятор NASM	8
3.3	Расширенный синтаксис командной строки NASM	9
3.4	Компоновщик LD	9
3.5	Запуск исполняемого файла	10
4	Задание для самостоятельной работы	11
5	Выводы	13
	Список литературы	14

Список иллюстраций

3.1	Создание каталога и файла	7
3.2	Открытие файла в текстовом редакторе	7
3.3	Ввод текста программы	8
3.4	Компиляция текста в объектный код	8
3.5	Компиляция файлов	9
3.6	Передача компоновщику	9
3.7	Создание исполняемого файла	9
3.8	Запуск файла	10
4.1	Копирование файла	11
4.2	Открытие файла	11
4.3	Изменение текста программы	11
4.4	Трансляция в объектный файл и компоновка файла	12
4.5	Запуск файла	12
4.6	Копирование файлов	12
4.7	Переход в каталог	12
4.8	Загрузка файлов на github	12

Список таблиц

1 Цель работы

Целью данной лабораторной работы является освоение процедуры компиляции и сборки программ, написанных на ассемблере NASM # Задание

Написать первые программы на языке NASM

2 Теоретическое введение

Основными функциональными элементами любой электронно-вычислительной машины (ЭВМ) являются центральный процессор, память и периферийные устройства. Взаимодействие этих устройств осуществляется через общую шину, к которой они подключены. Физически шина представляет собой большое количество проводников, соединяющих устройства друг с другом. В современных компьютерах проводники выполнены в виде электропроводящих дорожек на материнской (системной) плате.

3 Выполнение лабораторной работы

3.1 Программа Hello world!

Создам каталог lab05 для работы с программами на языке ассемблера NASM(рис. 3.1)

Перейду в созданный каталог (рис. 3.1)

Создам текстовый файл с именем hello.asm (рис. 3.1) и открою этот файл с помощью текстового редактора gedit (рис. 3.2)

```
[aavershinina@fedora arch-pc]$ mkdir lab05  
[aavershinina@fedora arch-pc]$ cd lab05  
[aavershinina@fedora lab05]$ touch hello.asm  
[aavershinina@fedora lab05]$ gedit hello.asm
```

Рис. 3.1: Создание каталога и файла



Рис. 3.2: Открытие файла в текстовом редакторе

Введу в него текст программы, выводящий текст *Hello, World!* (рис. 3.3)



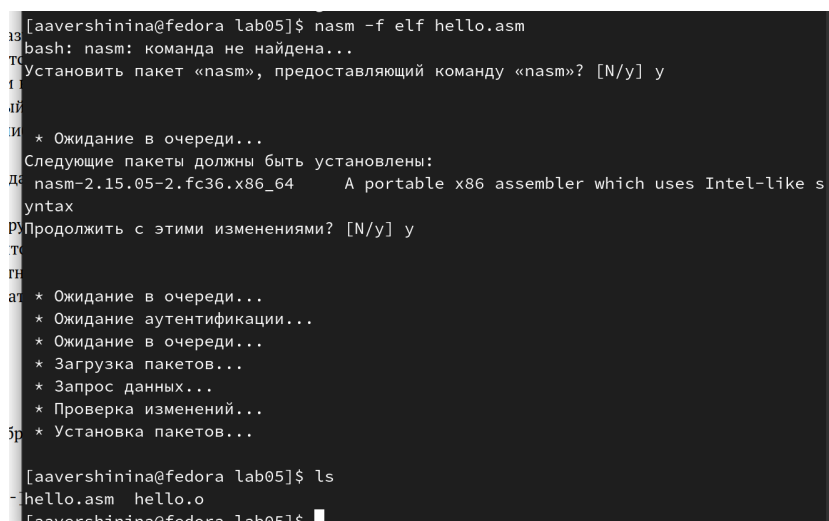
```
1 ; hello.asm
2 SECTION .data ; Начало секции данных
3 hello: DB 'Hello world!',10 ; 'Hello world!' плюс
4 ; символ перевода строки
5 helloLen: EQU $-hello ; Длина строки hello
6 SECTION .text ; Начало секции кода
7 GLOBAL _start
8 _start: ; Точка входа в программу
9 mov eax,4 ; Системный вызов для записи (sys_write)
10 mov ebx,1 ; 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. 3.3: Ввод текста программы

3.2 Транслятор NASM

Скомпилирую текст программы в объектный код при помощи команды (рис. 3.4)

С помощью команды `ls` проверю, что объектный файл был создан. Объектный файл имеет имя *hello.o* (рис. 3.4)



```
[aavershinina@fedora lab05]$ nasm -f elf hello.asm
bash: nasm: команда не найдена...
Установить пакет «nasm», предоставляющий команду «nasm»? [N/y] y
* Ожидание в очереди...
Следующие пакеты должны быть установлены:
nasm-2.15.05-2.fc36.x86_64 A portable x86 assembler which uses Intel-like syntax
Продолжить с этими изменениями? [N/y] y
* Ожидание в очереди...
* Ожидание аутентификации...
* Ожидание в очереди...
* Загрузка пакетов...
* Запрос данных...
* Проверка изменений...
* Установка пакетов...

[aavershinina@fedora lab05]$ ls
hello.asm hello.o
[aavershinina@fedora lab05]$
```

Рис. 3.4: Компиляция текста в объектный код

3.3 Расширенный синтаксис командной строки NASM

Выполню команду, которая скомпилирует исходный файл `hello.asm` в `obj.o` (опция `-o` позволяет задать имя объектного файла, в данном случае `obj.o`), при этом формат выходного файла будет `elf`, и в него будут включены символы для отладки (опция `-g`), кроме того, будет создан файл листинга `list.lst` (опция `-l`). С помощью команды `ls` проверьте, что файлы были созданы. (рис. 3.5)

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

Рис. 3.5: Компиляция файлов

3.4 Компоновщик LD

Чтобы получить исполняемую программу, объектный файл передам на обработку компоновщику. С помощью команды `ls` проверю, что исполняемый файл `hello` был создан. (рис. 3.6)

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

Рис. 3.6: Передача компоновщику

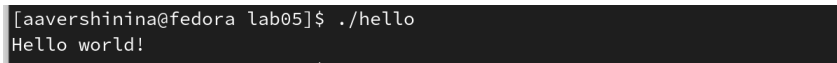
Выполню следующую команду (рис. 3.7). Имя объектного файла `obj.o` и имя исполняемого файла `main`.

```
[aavershinina@fedora lab05]$ ld -m elf_i386 obj.o -o main
[aavershinina@fedora lab05]$ ld --help
Использование ld [параметры] файл...
Параметры:
  -a КЛЮЧЕВОЕ СЛОВО                Управление общей библиотекой для совместимости с H
P/UX
  -A АРХИТЕКТУРА, --architecture АРХИТЕКТУРА  Задать архитектуру
  -b БИТОВЫЙ ФОРМАТ, --format БИТОВЫЙ ФОРМАТ  Задать битовый формат
```

Рис. 3.7: Создание исполняемого файла

3.5 Запуск исполняемого файла

Запущу на выполнение созданный исполняемый файл, находящийся в текущем каталоге(рис. 3.8)

A terminal window with a dark background. The prompt is [aavershinina@fedora lab05]\$. The command entered is ./hello. The output is Hello world!

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

Рис. 3.8: Запуск файла

4 Задание для самостоятельной работы

В каталоге `~/work/arch-pc/lab05` с помощью команды `cp` создам копию файла `hello.asm` с именем `lab5.asm` (рис. 4.1)

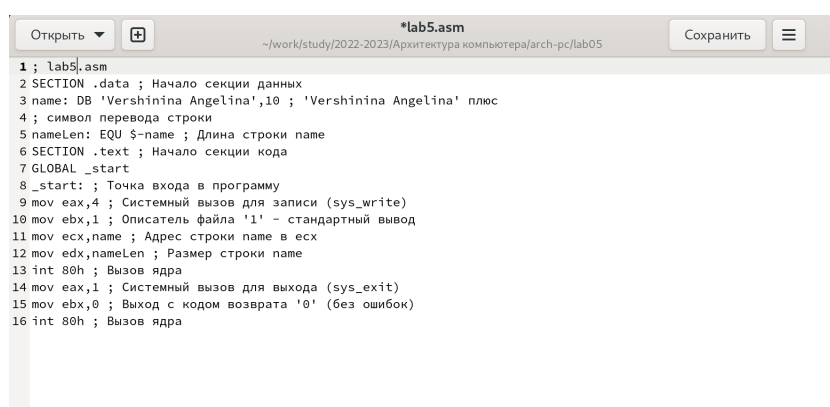
```
[aavershinina@fedora lab05]$ cp hello.asm lab5.asm
```

Рис. 4.1: Копирование файла

Открою файл с помощью текстового редактора (рис. 4.2) и внесу изменения в текст программы в файле `lab5.asm` так, чтобы вместо `Hello world!` на экран выводилась строка с моими фамилией и именем. (рис. 4.3)

```
[aavershinina@fedora lab05]$ gedit lab5.asm
```

Рис. 4.2: Открытие файла



```
*lab5.asm
1 ; lab5.asm
2 SECTION .data ; Начало секции данных
3 name: DB 'Vershina Angelina',10 ; 'Vershina Angelina' плюс
4 ; символ перевода строки
5 nameLen: EQU $-name ; Длина строки name
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,name ; Адрес строки name в ecx
12 mov edx,nameLen ; Размер строки name
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: Изменение текста программы

Оттранслирую полученный текст программы `lab5.asm` в объектный файл.

Выполню компоновку объектного файла и запущу получившийся исполняемый файл. (рис. 4.4 и 4.5)

```
[aavershinina@fedora lab05]$ nasm -f elf hello.asm
[aavershinina@fedora lab05]$ nasm -f elf lab5.asm
[aavershinina@fedora lab05]$ ls
hello  hello.asm  hello.o  lab5.asm  lab5.o  list.lst  main  obj.o
[aavershinina@fedora lab05]$ nasm -o obj.o -f elf -g -l list.lst lab5.asm
[aavershinina@fedora lab05]$ ld -m elf_i386 lab5.o -o lab5
```

Рис. 4.4: Трансляция в объектный файл и компоновка файла

```
[aavershinina@fedora lab05]$ ld -m elf_i386 lab5.o -o lab5
[aavershinina@fedora lab05]$ ./lab5
Vershinina Angelina
[aavershinina@fedora lab05]$
```

Рис. 4.5: Запуск файла

Скопирую файлы hello.asm и lab5.asm в мой локальный репозиторий в каталог ~/work/study/2022-2023/“Архитектура компьютера”/arch- pc/labs/lab05/.(рис. 4.6) Перейду в этот каталог.(рис. 4.7)

```
[aavershinina@fedora lab05]$ cp hello.asm lab5.asm ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc/labs/lab05/
[aavershinina@fedora lab05]$
```

Рис. 4.6: Копирование файлов

```
[aavershinina@fedora lab05]$ cd ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc/labs/lab05/
[aavershinina@fedora lab05]$
```

Рис. 4.7: Переход в каталог

Загружу файлы на github (рис. 4.8)

```
arch-pc/labs/lab05/
[aavershinina@fedora lab05]$ git add .
[aavershinina@fedora lab05]$ git commit -am 'feat(main): add files lab-5'
[master 7e6d554] feat(main): add files lab-5
2 files changed, 32 insertions(+)
create mode 100644 labs/lab05/hello.asm
create mode 100644 labs/lab05/lab5.asm
```

Рис. 4.8: Загрузка файлов на github

5 Выводы

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

Список литературы