

Лабораторная работа 4

Язык ассамблера NASM

Ромицына Анастасия Романовна НПИбд-02-23

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	12

Список иллюстраций

3.1	Создаие каталога	7
3.2	Переход в каталог с помощью cd	7
3.3	Создание текстового файла	7
3.4	Открытие текстового файла	7
3.5	Редактирование текстового файла	8
3.6	Компиляция текста	8
3.7	Компиляция, проверка	9
3.8	Пеедача на обработку	9
3.9	Создание исполняемого файла	9
3.10	Запуск программы	9
3.11	Запуск программы	10
3.12	Вносим изменения в программу	10
3.13	Запуск новой программы	10
3.14	Копирование в личный репозиторий	11

Список таблиц

1 Цель работы

Освоение процедуры компиляции и сборки программ, написанных на ассамблере NASM

2 Задание

Скомпилировать файлы и загрузить их на githab

3 Выполнение лабораторной работы

1.1 Программа Hello world! Рассмотрим пример простой программы на языке ассемблера NASM. Традиционно первая программа выводит приветственное сообщение Hello world! на экран. Создадим каталог для работы с программами на языке ассемблера NASM:(рис. 3.1)

```
[romitsinaar@fedora ~]$ mkdir -p ~/work/arch-pc/lab04
```

Рис. 3.1: Создаие каталога

1.2 Перейдите в созданный каталог(рис. 3.2)

```
[romitsinaar@fedora ~]$ cd ~/work/arch-pc/lab04
```

Рис. 3.2: Переход в каталог с помощью cd

1.3 Создаем текстовый файл с именем hello.asm(рис. 3.3)

```
[romitsinaar@fedora lab04]$ touch hello.asm  
[romitsinaar@fedora lab04]$ ls  
hello.asm
```

Рис. 3.3: Создание текстового файла

1.4 Открываем этот файл с помощью любого текстового редактора, например, gedit(рис. 3.4)

```
hello.asm  
[romitsinaar@fedora lab04]$ gedit hello.asm
```

Рис. 3.4: Открытие текстового файла

1.5 Вводим в него следующий текст:(рис. 3.5)

```
1 ; hello.asm
2 SECTION .data
3     hello: DB 'Hello world!',10
4     helloLen: EQU $-hello
5 SECTION .text
6     GLOBAL _start
7 _start:
8     mov eax,4
9     mov ebx,1
10    mov ecx,hello
11    mov edx,helloLen
12    int 80h
13
14    mov eax,1
15    mov ebx,0
16    int 80h |
```

Рис. 3.5: Редактирование текстового файла

2 NASM превращает текст программы в объектный код. Например, для компиляции приведённого выше текста программы «Hello World» необходимо написать:(рис. 3.6)

```
[romitsinaar@fedora lab04]$ nasm -f elf hello.asm
[romitsinaar@fedora lab04]$ ls
hello.asm  hello.o
```

Рис. 3.6: Компиляция текста

3 Выполняем следующую команду, Данная команда 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. 3.7)


```
[romitsinaar@fedora lab04]$ nasm -o obj.o -f elf -g -l list.lst hello.asm
[romitsinaar@fedora lab04]$ ls
hello.asm  hello.o  list.lst  obj.o
```

Рис. 3.7: Компиляция, проверка

4.1 чтобы получить исполняемую программу, объектный файл необходимо передать на обработку компоновщику:(рис. 3.8)

```
[romitsinaar@fedora lab04]$ ld -m elf_i386 hello.o -o hello
[romitsinaar@fedora lab04]$ ls
hello  hello.asm  hello.o  list.lst  obj.o
```

Рис. 3.8: Передача на обработку

4.2 Ключ -o с последующим значением задаёт в данном случае имя создаваемого исполняемого файла. Выполняем следующую команду:(рис. 3.9)

```
[romitsinaar@fedora lab04]$ ld -m elf_i386 obj.o -o main
```

Рис. 3.9: Создание исполняемого файла

5 Запустить на выполнение созданный исполняемый файл, находящийся в текущем каталоге, можно, набрав в командной строке (рис. 3.10)

```
[romitsinaar@fedora ~]$ cd ~/work/arch-pc/lab04
[romitsinaar@fedora lab04]$ ./hello
Hello world!
[romitsinaar@fedora lab04]$ cp hello.asm lab4.asm
[romitsinaar@fedora lab04]$ gedit lab4.asm
```

Рис. 3.10: Запуск программы

6.1 В каталоге ~/work/arch-pc/lab04 с помощью команды cp создайте копию файла hello.asm с именем lab4.asm(рис. 3.11)

```
[romitsinaar@fedora lab04]$ cp hello.asm lab4.asm
[romitsinaar@fedora lab04]$ gedit lab4.asm
```

Рис. 3.11: Запуск программы

6.2 С помощью любого текстового редактора вносим изменения в текст программы в файле lab4.asm так, чтобы вместо Hello world! на экран выводилась строка с моим фамилией и именем.(рис. 3.12)

```
1 ; hello.asm
2 SECTION .data
3     hello: DB 'Ромицына Анастасия',10
4     helloLen: EQU $-hello
5 SECTION .text
6     GLOBAL _start
7 _start:
8     mov eax,4
9     mov ebx,1
10    mov ecx,hello
11    mov edx,helloLen
12    int 80h
13
14    mov eax,1
15    mov ebx,0
16    int 80h
```

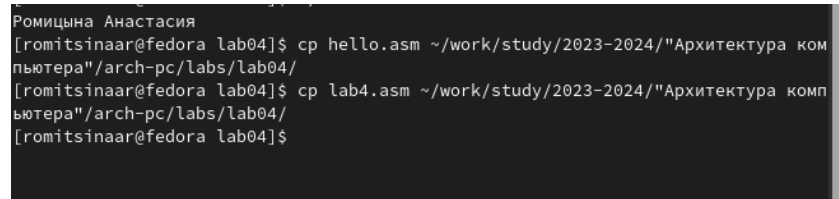
Рис. 3.12: Вносим изменения в программу

6.3 Оттранслируем полученный текст программы lab4.asm в объектный файл. Выполним компоновку объектного файла и запустим получившийся исполняемый файл.(рис. 3.13)

```
< [romitsinaar@fedora lab04]$ ./hello
Ромицына Анастасия
[romitsinaar@fedora lab04]$
```

Рис. 3.13: Запуск новой программы

6.4 Скопируем файлы `hello.asm` и `lab4.asm` в Ваш локальный репозиторий в каталог `~/work/study/2023-2024/“Архитектура компьютера”/arch-pc/labs/lab04/`. (рис. 3.14)



```
Ромицына Анастасия
[romitsinaar@fedora lab04]$ cp hello.asm ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab04/
[romitsinaar@fedora lab04]$ cp lab4.asm ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab04/
[romitsinaar@fedora lab04]$
```

Рис. 3.14: Копирование в личный репозиторий

4 Выводы

Мы смогли освоить процедуры компиляции и сборки программ, написанных на ассамблере NASM.