

# **Лабораторная работа №4**

**Создание и процесс обработки программ на языке ассемблера NASM**

Семенов Сергей Алексеевич

# Содержание

1	Цель работы	5
2	Выводы	10

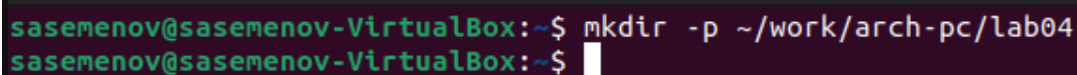
# Список иллюстраций

1.1	Создание каталога . . . . .	5
1.2	Открытие созданного каталога . . . . .	5
1.3	Создание текстового файла . . . . .	5
1.4	Открытие созданного файла и ввод текста . . . . .	6
1.5	Команда для компиляции текста программы . . . . .	6
1.6	Проверка работы команды . . . . .	6
1.7	Компиляция исходного файла . . . . .	7
1.8	Проверка . . . . .	7
1.9	Передача на обработку компановщику . . . . .	7
1.10	Проверка . . . . .	7
1.11	Выполнение новой команды . . . . .	7
1.12	Проверка . . . . .	8
1.13	Запуск на выполнение созданного исполняемого файла . . . . .	8
1.14	Создание копии файла . . . . .	8
1.15	Изменение текста программы . . . . .	9
1.16	Трансляция текста программы . . . . .	9

## **Список таблиц**

# 1 Цель работы

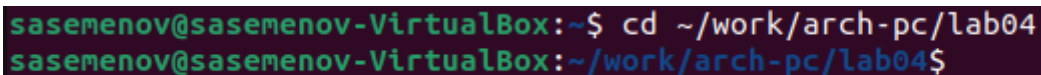
Целью работы является освоение процедуры компиляции и сборки программ, написанных на ассемблере NASM. # Выполнение лабораторной работы Создадим каталог для работы с программами на языке ассемблера NASM

A terminal window with a dark background. The prompt is 'sasemenov@sasemenov-VirtualBox:~\$'. The command 'mkdir -p ~/work/arch-pc/lab04' is entered and executed. The prompt changes to 'sasemenov@sasemenov-VirtualBox:~\$' with a cursor at the end.

```
sasemenov@sasemenov-VirtualBox:~$ mkdir -p ~/work/arch-pc/lab04
sasemenov@sasemenov-VirtualBox:~$
```

Рис. 1.1: Создание каталога

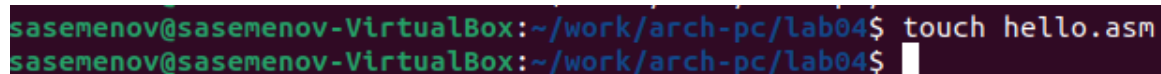
Перейдём в созданный каталог.

A terminal window with a dark background. The prompt is 'sasemenov@sasemenov-VirtualBox:~\$'. The command 'cd ~/work/arch-pc/lab04' is entered and executed. The prompt changes to 'sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04\$'.

```
sasemenov@sasemenov-VirtualBox:~$ cd ~/work/arch-pc/lab04
sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04$
```

Рис. 1.2: Открытие созданного каталога

Создадим текстовый файл с именем hello.asm.

A terminal window with a dark background. The prompt is 'sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04\$'. The command 'touch hello.asm' is entered and executed. The prompt changes to 'sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04\$' with a cursor at the end.

```
sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04$ touch hello.asm
sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04$
```

Рис. 1.3: Создание текстового файла

Откроем этот файл с помощью текстового редактора gedit и введём в него текст из лабораторной работы.

текстового редактора, например, gedit

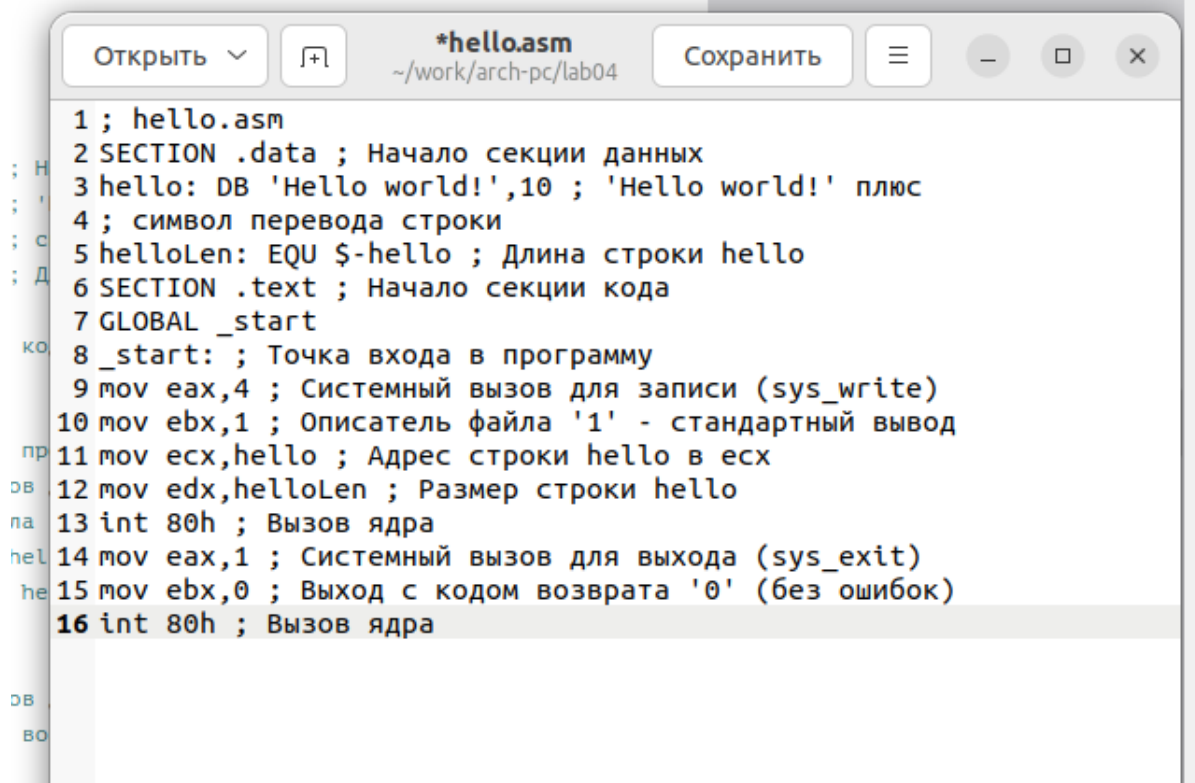


Рис. 1.4: Открытие созданного файла и ввод текста

Для компиляции текста программы «Hello World» необходимо написать:

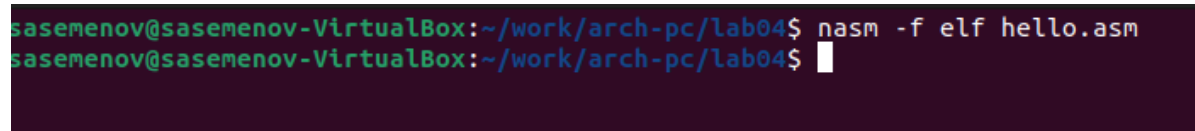


Рис. 1.5: Команда для компиляции текста программы

Проверим работу команды

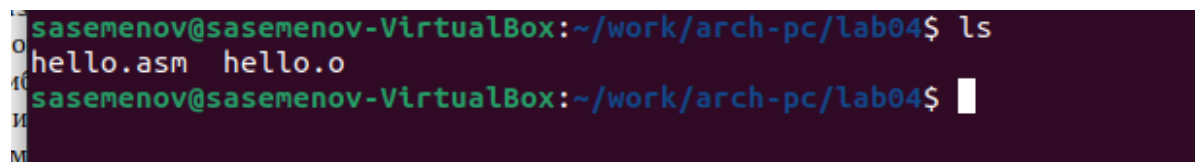


Рис. 1.6: Проверка работы команды

Скомпилируем исходный файл hello.asm в obj.o.

```
sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.lst hello.asm
```

Рис. 1.7: Компиляция исходного файла

С помощью команды ls проверим, что файлы были созданы.

```
sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04$ ls  
hello hello.asm hello.o list.lst main obj.o
```

Рис. 1.8: Проверка

Получим исполняемую программу и объектный файл. Для этого передадим на обработку компоновщику.

```
sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04$ ld -m elf_i386 hello.o -o hello
```

Рис. 1.9: Передача на обработку компоновщику

С помощью команды ls проверим, что исполняемый файл hello был создан.

```
sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04$ ls  
hello hello.asm hello.o list.lst main obj.o
```

Рис. 1.10: Проверка

Теперь выполним следующую команду:

```
sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04$ ld -m elf_i386 obj.o -o main
```

Рис. 1.11: Выполнение новой команды

С помощью команды ls выполним проверку.

```
sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04$ ls
hello  hello.asm  hello.o  list.lst  main  obj.o
```

Рис. 1.12: Проверка

Запустим на выполнение созданный исполняемый файл, находящийся в текущем каталоге.

```
sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04$ ./hello
Hello world!
```

Рис. 1.13: Запуск на выполнение созданного исполняемого файла

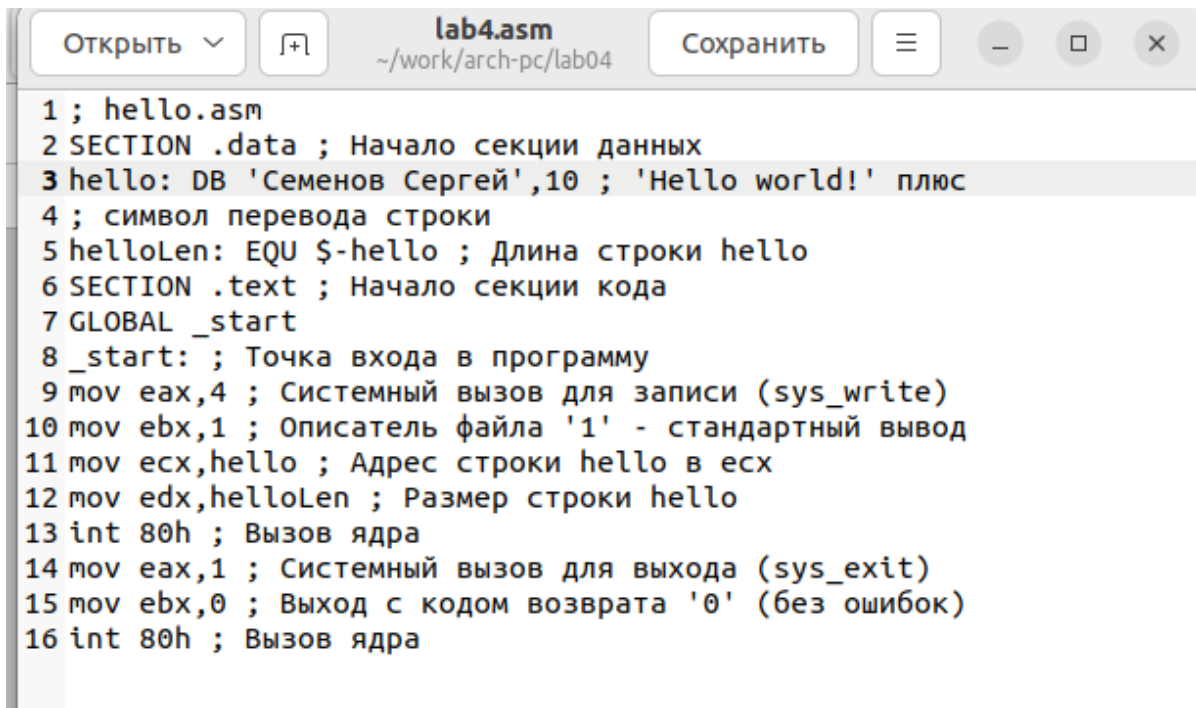
В каталоге ~/work/arch-pc/lab04 с помощью команды `cp` создадим копию файла `hello.asm` с именем `lab4.asm`.

```
sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04$ cp hello.asm lab4.asm
sasemenov@sasemenov-VirtualBox:~/work/arch-pc/lab04$
```

Рис. 1.14: Создание копии файла

С помощью текстового редактора внесём изменения в текст программы в файле `lab4.asm` так, чтобы вместо `Hello world!` на экран выводилась строка с фамилией и именем.

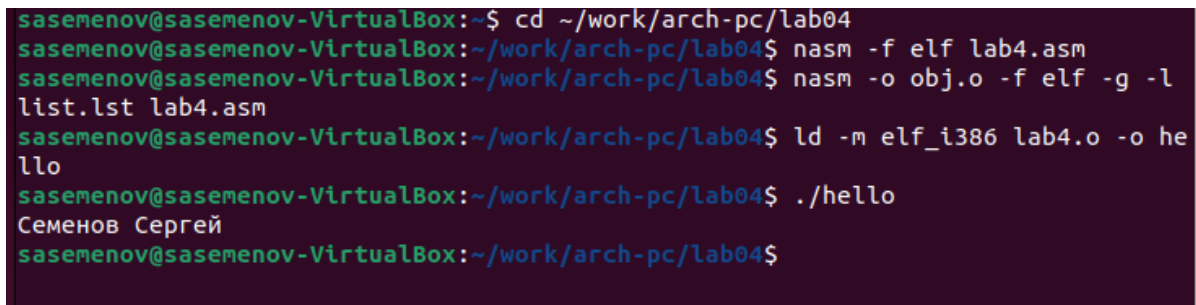




```
1 ; hello.asm
2 SECTION .data ; Начало секции данных
3 hello: DB 'Семенов Сергей',10 ; 'Hello world!' плюс
4 ; символ перевода строки
5 helloLen: EQU $-hello ; Длина строки hello
6 SECTION .text ; Начало секции кода
7 GLOBAL _start
8 _start: ; Точка входа в программу
9 mov eax,4 ; Системный вызов 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. 1.15: Изменение текста программы

Оттранслируем полученный текст программы lab4.asm в объектный файл. Выполним компоновку объектного файла и запустим получившийся исполняемый файл.



```
sasemenov@sasemenov-VirtualBox:~$ cd ~/.work/arch-pc/lab04
sasemenov@sasemenov-VirtualBox:~/.work/arch-pc/lab04$ nasm -f elf lab4.asm
sasemenov@sasemenov-VirtualBox:~/.work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l
list.lst lab4.asm
sasemenov@sasemenov-VirtualBox:~/.work/arch-pc/lab04$ ld -m elf_i386 lab4.o -o he
llo
sasemenov@sasemenov-VirtualBox:~/.work/arch-pc/lab04$ ./hello
Семенов Сергей
sasemenov@sasemenov-VirtualBox:~/.work/arch-pc/lab04$
```

Рис. 1.16: Трансляция текста программы

## 2 Выводы

В ходе выполнения лабораторной работы мы освоили процедуры компиляции и сборки программ, написанных на ассемблере NASM.