

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 4

дисциплина: Архитектура компьютера

Студент: Лебеденко Елена Викторовна

Группа: НКАбд-04-25

МОСКВА

2025 г.

Содержание

1. Цель работы.....	4
2. Выполнение лабораторной работы.....	5-7
2.1 Программа Hello world!.....	5
2.2 Транслятор NASM.....	5
2.3 Расширенный синтаксис командной строки NASM.....	6
2.4 Компоновщик LD.....	6
2.5 Запуск исполняемого файла.....	7
3. Задания для самостоятельной работы.....	7-8
4. Вывод.....	9

Список иллюстраций

1.1 Создание каталога.....	5
1.2Создание текстового файла.....	5
1.3 Текстовый редактор.....	5
1.4 Компиляция программы.....	6
1.5 Проверка.....	6
1.6 Синтаксис NASM.....	6
1.7 Обработка файла.....	6
1.8 Создание исполняемого файла.....	6
1.9 Запуск программы.....	7
2.1 Задание 1.....	7
2.2 Задание 2 (1).....	7
2.3 Задание 2 (2).....	7
2.4 Задание 3	8
2.5 Задание 4.....	8

1. Цель работы

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

2. Выполнение лабораторной работы

2.1. Программа Hello world!

1) Создаю каталог для работы с программами на языке ассемблера NASM и перехожу в него.

```
evlebedenko@evlebedenko:~$ mkdir -p ~/work/arch-pc/lab04
evlebedenko@evlebedenko:~$ cd ~/work/arch-pc/lab04
```

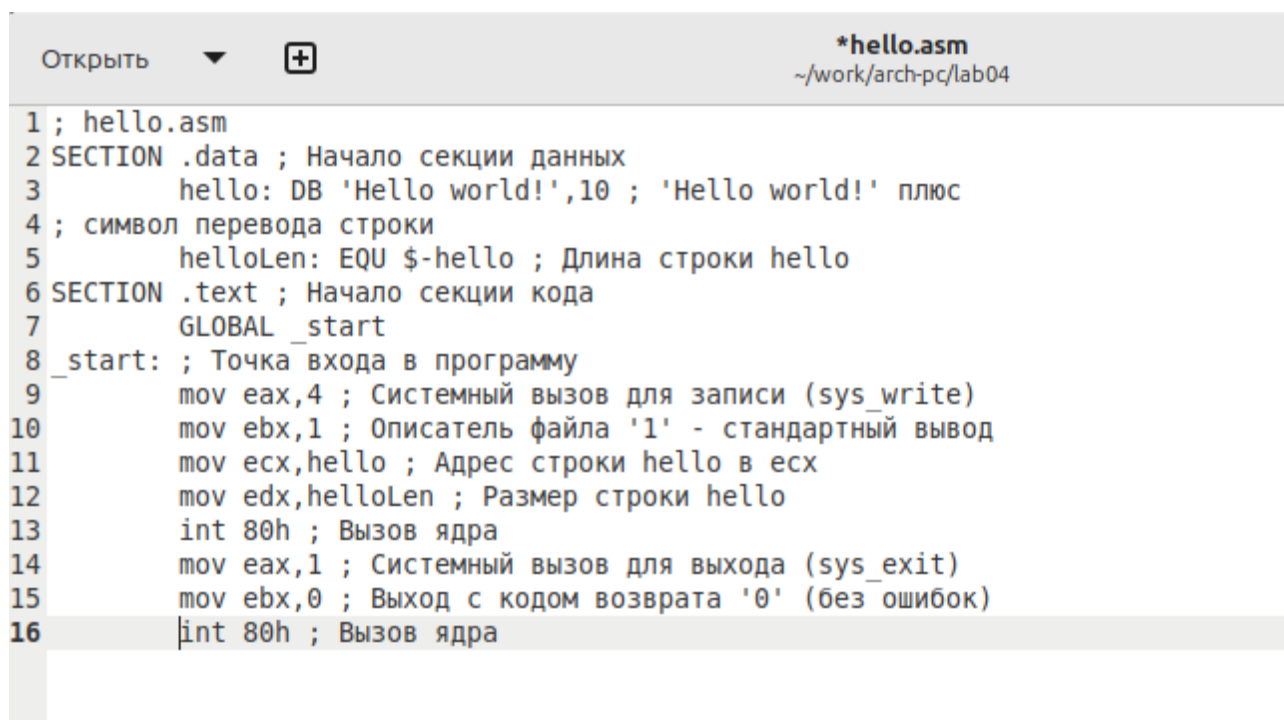
1.1 Создание каталога

2) Создаю текстовый файл с именем hello.asm, открываю этот файл с помощью текстового редактора gedit.

```
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ touch hello.asm
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ gedit hello.asm
```

1.2 Создание текстового файла

3) Ввожу данный в задании текст.



```
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 есх
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
```

1.3 Текстовый редактор

2.2 Транслятор NASM

1) Преобразую текст программы из файла hello.asm в объектный код, который запишется в файл hello.o.

```
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ nasm -f elf hello.asm
```

1.4 Компиляция программы

```
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ ls  
hello.asm hello.o
```

1.5 Проверка

2.3 Расширенный синтаксис командной строки NASM

1) Выполняю команду, которая скомпилирует исходный файл hello.asm в obj.o (опция -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). Проверю выполнение команды с помощью ls.

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

1.6 Синтаксис NASM

2.4. Компоновщик LD

1) Передаю объектный файл на обработку компоновщику. С помощью команды ls проверяю что исполняемый файл hello был создан.

```
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ ld -m elf_i386 hello.o -o hello  
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ ls  
hello hello.asm hello.o list.lst obj.o
```

1.7 Обработка файла

2) Выполняю команду из задания.

```
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ ld -m elf_i386 obj.o -o main  
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ ls  
hello hello.asm hello.o list.lst main obj.o
```

1.8 Создание исполняемого файла

Исполняемый файл имеет имя main, он собран из файла obj.o .

2.5. Запуск исполняемого файла

1) Запускаю исполняемый файл с помощью данной команды.

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

1.9 Запуск программы

3. Задание для самостоятельной работы

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

```
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ cp ~/work/arch-pc/lab04/hello.asm ~/work/arch-pc/lab04/lab4.asm
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ ls
hello  hello.asm  hello.o  lab4.asm  list.lst  main  obj.o
```

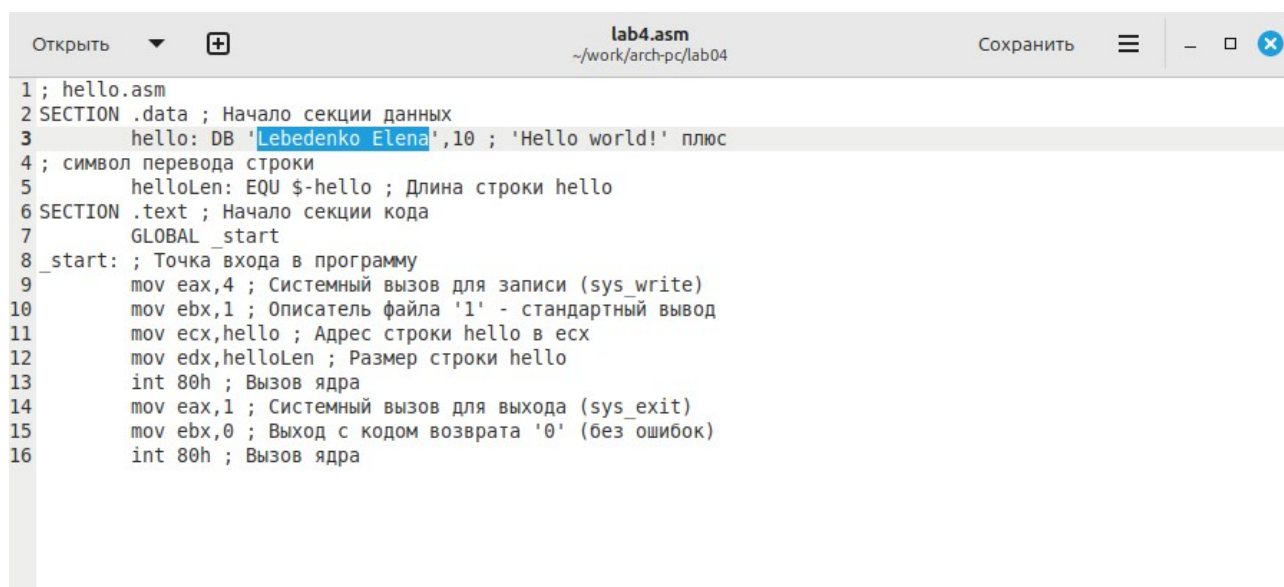
2.1 Задание 1

Создаю копию файла hello.asm с именем lab4.asm.

2) С помощью любого текстового редактора внесите изменения в текст программы в файле lab4.asm так, чтобы вместо Hello world! на экран выводилась строка с вашими фамилией и именем.

```
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ gedit lab4.asm
```

2.2 Задание 2 (1)



```
lab4.asm
~/work/arch-pc/lab04

1; hello.asm
2 SECTION .data ; Начало секции данных
3     hello: DB 'Lebedenko Elena',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
```

2.3 Задание 2 (2)

3) Оттранслируйте полученный текст программы lab4.asm в объектный файл. Выполните компоновку объектного файла и запустите получившийся исполняемый файл.

```
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ nasm -f elf lab4.asm
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ ld -m elf_i386 lab4.o -o lab4
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ ls
hello  hello.asm  hello.o  lab4  lab4.asm  lab4.o  list.lst  main  obj.o
evlebedenko@evlebedenko:~/work/arch-pc/lab04$ ./lab4
Lebedenko Elena
```

2.4 Задание 3

4) Скопируйте файлы hello.asm и lab4.asm в Ваш локальный репозиторий в каталог ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab04/.

Загрузите файлы на Github.

```
evlebedenko@evlebedenko:~/work/study/2025-2026/Архитектура компьютера/arch-pc$ git add .
evlebedenko@evlebedenko:~/work/study/2025-2026/Архитектура компьютера/arch-pc$ git commit
-am 'feat(main): make course structure'
[master f118c94] feat(main): make course structure
 2 files changed, 32 insertions(+)
 create mode 100644 labs/lab04/hello.asm
 create mode 100644 labs/lab04/lab4.asm
evlebedenko@evlebedenko:~/work/study/2025-2026/Архитектура компьютера/arch-pc$ git push
Перечисление объектов: 9, готово.
Подсчет объектов: 100% (9/9), готово.
При сжатии изменений используется до 2 потоков
Сжатие объектов: 100% (6/6), готово.
Запись объектов: 100% (6/6), 948 байтов | 948.00 КиБ/с, готово.
Всего 6 (изменений 3), повторно использовано 0 (изменений 0), повторно использовано пакетов
в 0
remote: Resolving deltas: 100% (3/3), completed with 2 local objects.
To github.com:Lebeduha21/study_2025-2026_arh-pc.git
 99d0f52..f118c94  master -> master
```

2.5 Задание 4

4. Вывод

При выполнении лабораторной работы я освоила базовые процедуры создание и компиляции программ на ассемблере NASM.