

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: Клименко Кирилл Русланович

Группа: НММбд-02-24

МОСКВА

2025г.

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	12
6	Список литературы	13

Список иллюстраций

4.1	Создание каталога	8
4.2	Переход в каталог	8
4.3	Создание текстового файла	8
4.4	Открытие файла	8
4.5	Ввод текста	9
4.6	Компиляция текста	9
4.7	Проверка	9
4.8	Создание файлов	9
4.9	Проверка	9
4.10	Передача файла на компоновку.....	10
4.11	Проверка, что исполняемый файл hello был создан.....	10
4.12	Зададим имя создаваемого исполняемого файла.....	10
4.13	Запуск на выполнение созданный исполняемый файл.....	10
4.14	Создание копии файла с именем lab4.asm.....	10
4.15	Внесение изменения в текст программы	11
4.16	Оттранслирование, компоновка, запуск	11
4.17	Копирование файлов в локальный репозиторий.....	11

Список таблиц

3.1	Описание некоторых каталогов файловой системы GNU Linux . . .	7
-----	---	---

1 Цель работы

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

2 Задание

1. Программа Hello world!

1.1 создать каталог для работы с программами на языке ассемблера NASM

1.2 перейти в созданный каталог

1.3 создать текстовый файл с именем hello.asm

1.4 открыть этот файл

1.5 ввести в него указанный текст

2. Транслятор Nasm

2.1 выполнить компиляцию в объектный код

3. Расширенный синтаксис

3.1 выполнить компиляцию исходного файла

4. Компоновщик LD

4.1 передать объектный файл на обработку компоновщику

5. Запустить исполняемый файл

6. Задания для самостоятельной работы

6.1 создать копию файла hello.asm с именем lab4.asm

6.2 изменить скопированный файл, чтобы выводилась строка с именем и фамилией

6.3 оттранслировать полученный текст программы lab4.asm в объектный файл

6.4 скопировать файлы hello.asm и lab4.asm в локальный репозиторий

3 Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы.

Например, в табл. 3.1 приведено краткое описание стандартных каталогов Unix.

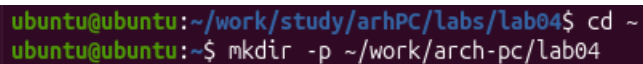
Таблица 3.1: Описание некоторых каталогов файловой системы GNU Linux

Имя каталога	Описание каталога
/	Корневая директория, содержащая всю файловую
/bin	Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям
/etc	Общесистемные конфигурационные файлы и файлы конфигурации установленных программ
/home	Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя
/media	Точки монтирования для сменных носителей
/root	Домашняя директория пользователя root
/tmp	Временные файлы
/usr	Вторичная иерархия для данных пользователя

Более подробно про Unix см. в [1–4].

4 Выполнение лабораторной работы

1. Создайте каталог для работы с программами на языке ассемблера NASM

A terminal window with a dark background. The prompt is 'ubuntu@ubuntu:~/work/study/archPC/labs/lab04\$'. The user enters 'cd ~'. The prompt changes to 'ubuntu@ubuntu:~\$'. The user enters 'mkdir -p ~/work/arch-pc/lab04'.

```
ubuntu@ubuntu:~/work/study/archPC/labs/lab04$ cd ~
ubuntu@ubuntu:~$ mkdir -p ~/work/arch-pc/lab04
```

Рис. 4.1: Создание каталога

2. Перейдём в созданный каталог

A terminal window with a dark background. The prompt is 'ubuntu@ubuntu:~\$'. The user enters 'cd ~/work/arch-pc/lab04'. The prompt changes to 'ubuntu@ubuntu:~/work/arch-pc/lab04\$' with a cursor.

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

Рис. 4.2: Переход в каталог

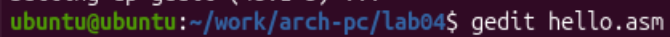
3. Создадим текстовый файл с именем hello.asm

A terminal window with a dark background. The prompt is 'ubuntu@ubuntu:~/work/arch-pc/lab04\$'. The user enters 'touch hello.asm'.

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

Рис. 4.3: Создание текстового файла

4. Откроем этот файл с помощью текстового редактора

A terminal window with a dark background. The prompt is 'ubuntu@ubuntu:~/work/arch-pc/lab04\$'. The user enters 'gedit hello.asm'.

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

Рис. 4.4: Открытие файла

5. Введём в него текст:

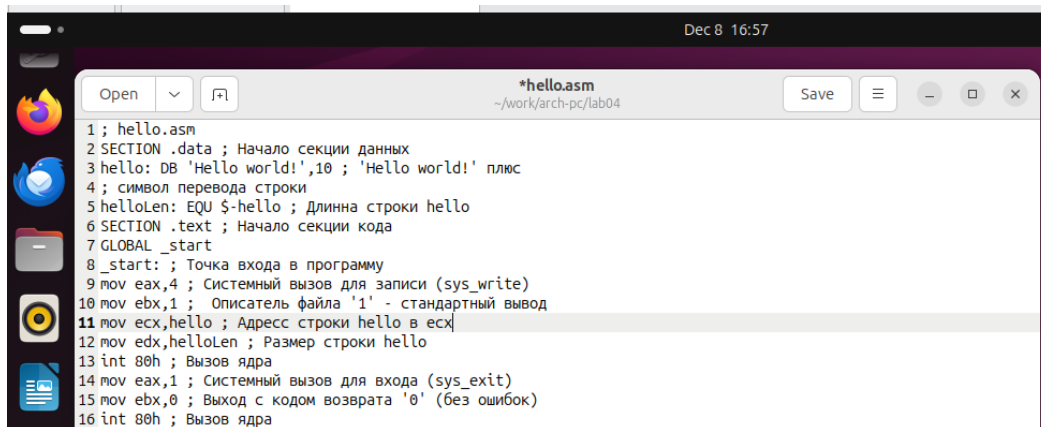


Рис. 4.5: Ввод текста

6. Скомпилируем данный текст

```

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

```

Рис. 4.6: Компиляция текста

7. Проверим, что объектный файл был создан:

```

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

```

Рис. 4.7: Проверка

8. Скомпилируем исходный файл hello.asm в obj.o и создадим файл листинга list.lst

```

ubuntu@ubuntu:~/work/arch-pc/lab04$ nasm -o obj.o -f elf -g -l list.lst hello.a
sm

```

Рис. 4.8: Создание файлов

9. Проверим, что файлы были созданы

```

ubuntu@ubuntu:~/work/arch-pc/lab04$ ls
hello.asm hello.o list.lst obj.o

```

Рис. 4.9: Проверка

10. Передаем объектный файл на обработку компоновщику для получения исполняемой программы

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

Рис. 4.10: Передача файла на компоновку

11. Проверим, что исполняемый файл hello был создан

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

Рис. 4.11: Проверка, что исполняемый файл hello был создан

12. Зададим имя создаваемого исполняемого файла

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

Рис. 4.12: Зададим имя создаваемого исполняемого файла

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

```
ubuntu@ubuntu:~/work/arch-pc/lab04$ ./main
Hello world!
```

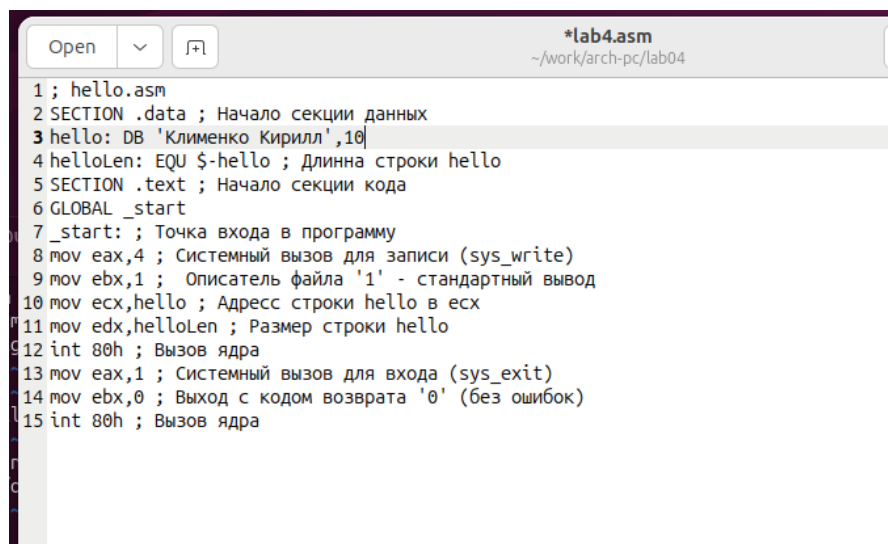
Рис. 4.13: Запуск на выполнение созданный исполняемый файл

14. Создадим копию файла hello.asm с именем lab4.asm

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

Рис. 4.14: Создание копии файла с именем lab4.asm

15. Внесём изменения в текст программы в файле lab4.asm

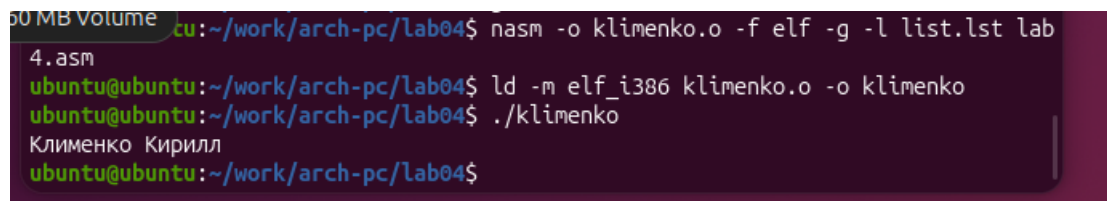


```
1; hello.asm
2 SECTION .data ; Начало секции данных
3 hello: DB 'Клименко Кирилл',10
4 helloLen: EQU $-hello ; Длина строки hello
5 SECTION .text ; Начало секции кода
6 GLOBAL _start
7 _start: ; Точка входа в программу
8 mov eax,4 ; Системный вызов для записи (sys_write)
9 mov ebx,1 ; Описатель файла '1' - стандартный вывод
10 mov ecx,hello ; Адрес строки hello в ecx
11 mov edx,helloLen ; Размер строки hello
12 int 80h ; Вызов ядра
13 mov eax,1 ; Системный вызов для входа (sys_exit)
14 mov ebx,0 ; Выход с кодом возврата '0' (без ошибок)
15 int 80h ; Вызов ядра
```

Рис. 4.15: Внесение изменения в текст программы

16. Оттранслируем полученный текст программы lab4.asm в объектный файл

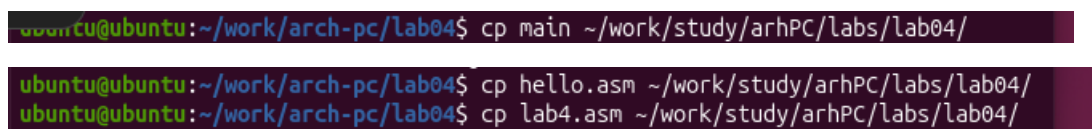
Выполним компоновку объектного файла и запустим получившийся исполняемый файл



```
ubuntu@ubuntu:~/work/arch-pc/lab04$ nasm -o klimenko.o -f elf -g -l list.lst lab4.asm
ubuntu@ubuntu:~/work/arch-pc/lab04$ ld -m elf_i386 klimenko.o -o klimenko
ubuntu@ubuntu:~/work/arch-pc/lab04$ ./klimenko
Клименко Кирилл
ubuntu@ubuntu:~/work/arch-pc/lab04$
```

Рис. 4.16: Оттранслирование, компоновка, запуск

17. Скопировал файлы hello.asm и lab4.asm в локальный репозиторий



```
ubuntu@ubuntu:~/work/arch-pc/lab04$ cp main ~/work/study/arhPC/labs/lab04/
ubuntu@ubuntu:~/work/arch-pc/lab04$ cp hello.asm ~/work/study/arhPC/labs/lab04/
ubuntu@ubuntu:~/work/arch-pc/lab04$ cp lab4.asm ~/work/study/arhPC/labs/lab04/
```

Рис. 4.17: Копирование файлов в локальный репозиторий

18. Загружаю файлы на Github

5 Выводы

В ходе выполнения работы, я освоил процедуры компиляции и сборки программ, написанных на ассемблере NASM

6 Список литературы

1. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.
2. Robbins A. Bash Pocket Reference. O'Reilly Media, 2016. 156 с.
3. Zarrelli G. Mastering Bash. Packt Publishing, 2017. 502 с.
4. Newham C. Learning the bash Shell: Unix Shell Programming. O'Reilly Media, 2005. 354 с.