

# **Лабораторная работа №4**

**Архитектура вычислительных систем**

Басманова Дарья Кириллова

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Задание</b>	<b>5</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
<b>4</b>	<b>Выполнение самостоятельной работы</b>	<b>8</b>
<b>5</b>	<b>Выводы</b>	<b>10</b>
	<b>Список литературы</b>	<b>11</b>

# Список иллюстраций

3.1	1.png	. . . . .	6
3.2	2.png	. . . . .	6
3.3	3.png	. . . . .	7
3.4	4.png	. . . . .	7
4.1	5.png	. . . . .	8
4.2	6.png	. . . . .	8
4.3	7.png	. . . . .	9
4.4	8.png	. . . . .	9

# 1 Цель работы

Целью работы является освоение процедуры оформления отчетов с помощью легковесного языка разметки Markdown.

## 2 Задание

1. В соответствующем каталоге сделайте отчёт по лабораторной работе №4 в формате Markdown. В качестве отчёта необходимо предоставить отчёты в 3 форматах: pdf, docx и md.
2. Загрузите файлы на github.

### 3 Выполнение лабораторной работы

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 3.1)

- 1) Рассмотрим пример простой программы на языке ассемблера NASM. Создадим каталог для работы с программами на языке ассемблера NASM.

```
dkbasmanova@dk5n55 ~ $ mkdir ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc/labs/lab04
```

Рис. 3.1: 1.png

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 3.2)

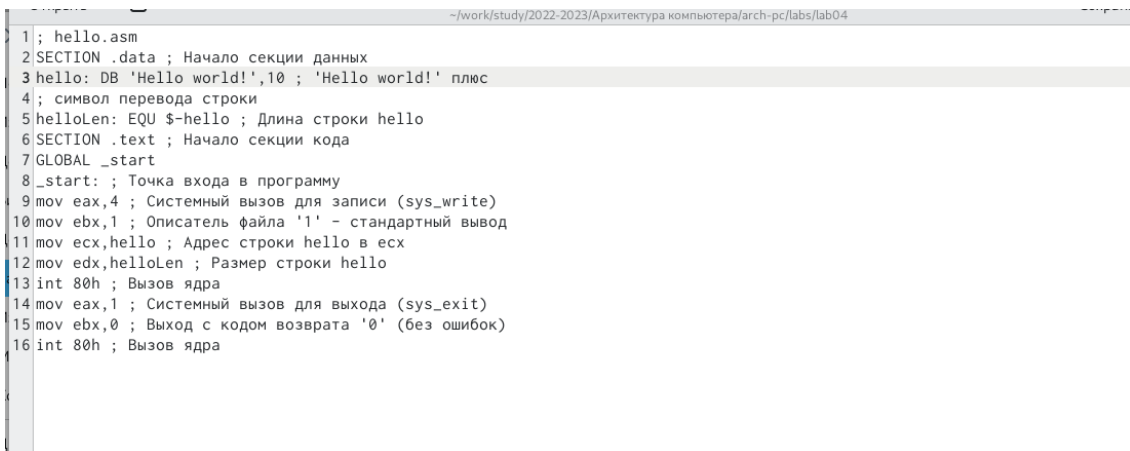
- 2) Перейдем в созданный каталог и создадим текстовый файл с именем hello.asm.

```
dkbasmanova@dk5n55 ~ $ cd ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc/labs/lab04
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ nasm -f elf hello.asm
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ gedit hello.asm
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ nasm -o obj.o -f elf -g -l list.lst hello.asm
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ls
hello.asm  hello.o  list.lst  obj.o  presentation  report
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ld -m elf_i386 hello.o -o hello
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ls
hello  hello.asm  hello.o  list.lst  obj.o  presentation  report
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ld -m elf_i386 obj.o -o main
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ld --help
Использование ld [параметры] файл...
Параметры:
  -a КЛЮЧЕВОЕ СЛОВО                Управление общей библиотекой для совместимости с HP/UX
  -A АРХИТЕКТУРА, --architecture АРХИТЕКТУРА  Задать архитектуру
  -b ЦЕЛЬ, --format ЦЕЛЬ            Задать цель для следующих входных файлов
  -c ФАЙЛ, --mri-script ФАЙЛ       Прочитать сценарий компоновщика в формате MRI
  -d, -dc, -dp                     Принудительно делать общие символы определёнными
```

Рис. 3.2: 2.png

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 3.3)

- 3) Откроем этот файл с помощью любого текстового редактора и введем следующий текст.

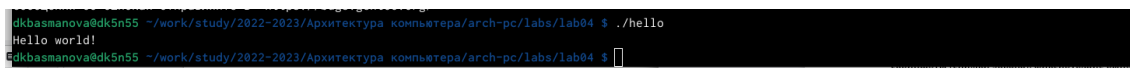


```
1 ; hello.asm
2 SECTION .data ; Начало секции данных
3 hello: DB 'Hello world!',10 ; 'Hello world!' плюс
4 ; символ перевода строки
5 helloLen: EQU $-hello ; Длина строки hello
6 SECTION .text ; Начало секции кода
7 GLOBAL _start
8 _start: ; Точка входа в программу
9 mov eax,4 ; Системный вызов для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

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 3.4)

- 4) NASM превращает текст программы в объектный код. Для компиляции приведённого выше текста программы «Hello World» напишем `nasm -f elf hello.asm`



```
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ./hello
Hello world!
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $
```

Рис. 3.4: 4.png

## 4 Выполнение самостоятельной работы

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 4.1)

- 1) В каталоге `~/work/arch-pc/lab05` с помощью команды `cp` создадим копию файла `hello.asm` с именем `lab5.asm`.

```
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ cp hello.asm lab05.asm
```

Рис. 4.1: 5.png

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 4.2)

- 2) С помощью любого текстового редактора внесем изменения в текст программы в файле `lab5.asm` так, чтобы вместо `Hello world!` на экран выводилась строка с фамилией и именем.

```
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ nasm -f elf -g -l list.lst lab05.asm
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ld -m elf_i386 lab05.o -o lab05
```

Рис. 4.2: 6.png

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 4.3)

- 3) Оттранслируем полученный текст программы `lab5.asm` в объектный файл и запустим.



```
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc/labs/lab04 $ ./lab05
Basmanova Daria
```

Рис. 4.3: 7.png

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 4.4)

4) Загружаем файлы на GitHub.

```
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ git add .
dkbasmanova@dk5n55 ~/work/study/2022-2023/Архитектура компьютера/arch-pc $ git commit -am 'feat(main): add files lab-4'
[master 81f7c8f] feat(main): add files lab-4
9 files changed, 49 insertions(+)
create mode 100755 labs/lab04/hello
create mode 100644 labs/lab04/hello.asm
create mode 100644 labs/lab04/hello.o
create mode 100755 labs/lab04/lab05
create mode 100644 labs/lab04/lab05.asm
create mode 100644 labs/lab04/lab05.o
create mode 100644 labs/lab04/list.lst
```

Рис. 4.4: 8.png

## 5 Выводы

Я освоила процедуру компиляции и сборки программ, написанных на ассемблере NASM.

## **Список литературы**