**РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ**

**Факультет физико-математических и естественных наук Кафедра прикладной информатики и теории вероятностей**

**ОТЧЕТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2.**

дисциплина: *Архитектура компьютеров*

Студент: Подхалюзина Виолетта Михайловна

Группа: НКАбд-04-24

**МОСКВА**

2024 г.

Содержание

1. [Цель работы 3](#_bookmark0)
2. [Введение 3](#_bookmark1)
3. [Выполнение лабораторной работы 4](#_bookmark2)
   1. [Установление необходимого ПО 4](#_bookmark3)

[Установка TexLive 4](#_bookmark4)

[Установка pandoc и pandoc-crossref 4](#_bookmark5)

* 1. [Заполнение отчета по выполнению лабораторной работы №4 с](#_bookmark6) [помощью языка разметки Markdown 4](#_bookmark6)
  2. [Задание для самостоятельной работы 5](#_bookmark7)
  3. [Контрольные вопросы для самопроверки 7](#_bookmark8)

1. [Вывод 8](#_bookmark9)
2. [Список литературы 9](#_bookmark10)

# Цель работы

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

# Введение

Markdown - легковесный язык разметки, созданный с целью обозначения форматирования в простом тексте, с максимальным сохранением его читаемости человеком, и пригодный для машинного преобразования в языки для продвинутых публикаций. Внутритекстовые формулы делаются аналогично формулам LaTeX. В Markdown вставить изображение в документ можно с помощью непосредственного указания адреса изображения. Синтаксис Markdown для встроенной ссылки состоит из части [link text], представляющей текст гиперссылки, и части (file-name.md) – URL-адреса или имени файла, на который дается ссылка. Markdown поддерживает как встраивание фрагментов кода в предложение, так и их размещение между предложениями в виде отдельных огражденных блоков. Огражденные блоки кода — это простой способ выделить синтаксис для фрагментов кода.

# Выполнение лабораторной работы

* 1. **Установление необходимого ПО** **Установка TexLive**

Сначала я установила TexLive на свой компьютер и распаковала необходимый архив при помощи функций cd открываем Загрузки, а далее используем ls и zcat.



Далее я перешла при помощи cd в нужную папку и запустила install-tl-\* с правами root и добавила /usr/local/texlive/2024/bin/x86\_64-linux.

Рис. 2: Запуск скриптаРис. 2: Запуск скрипта

# Установка pandoc и pandoc-crossref

Далее я скачала pandoc и архив pandoc-crossref.

Рис. 4: Скачивание pandocРис. 5: Скачивание pandoc-crossref

После этого мне было необходимо распаковать файлы и скопировать файлы в отдельный каталог при помощи sudo.

Рис. 7: Копирование каталогов в другую директориюРис. 7: Копирование каталогов в другую директорию

Далее при помощи lc проверила выполнились ли заданные команды.

Рис. 8: Проверка правильности выполнения команды

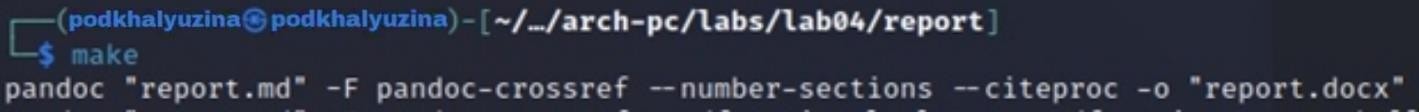
# Заполнение отчета по выполнению лабораторной работы №4 с помощью языка разметки Markdown

После этого я перешла в каталог курса из прошлой лабораторной работы при помощи функции cd: cd work/study/2023-2024/"Архитектура компьютера"

/arch-pc

Я получила изменения из удалённого репозитория и обновила локальную рабочую копию при помощи команды git pull.

Перешла в каталог с шаблоном отчета с помощью cd и ввела команду make, чтобы скомпилировать шаблон с использованием Makefile. Если запустить make, то программа попытается найти файл с именем по умолчание Makefile в текущем каталоге и выполнить инструкции из него.



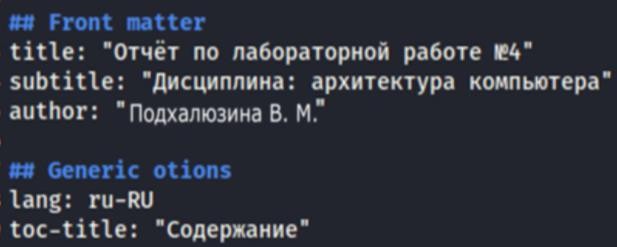
После этого я открыла сгенерированные файлы и убедилась, что они правильно были открыты и созданы.

Я с использованием Makefile и вводя команду make clean удалила результаты предыдущей сборки. Операция затрагивает файлы в каталоге исходников, файлы, установленные при помощи команды make install, затронуты не будут.

Открыла файл report.md с помощью mousepad – простейший текстовый редактор.



После этого я заполнила отчет с помощью языка разметки Markdown - он позволяет создавать тексты без использования Word и других редакторов.



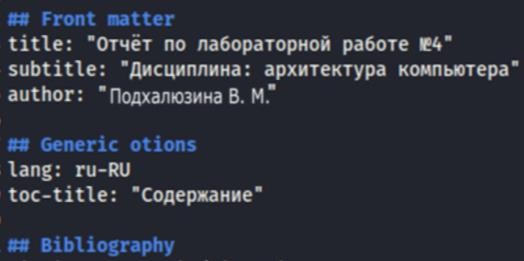
Далее я скомпилировала файл с отчетом и загрузила его на GitHub.

# Задание для самостоятельной работы

Я перешла в директорию lab03/report и заполнила там отчет по третьей лабораторной работе.

Рис. 19: Перемещение между директориями

Я скопировала файл report.md и с помощью текстового редактора mousepad и начинаю заполнила отчет.



Удалила предыдущий файл отчета при помощи rm — утилиты, используемой для безвозвратного удаления файлов из файловой системы.

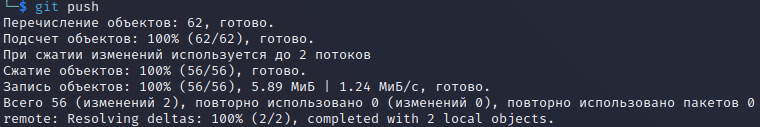
После чего я скомпилировала файл с отчетом при помощи make.

Рис. 23: Компиляция файлов

Далее я удалила лишние сгенерированные файлы report.docx и report.pdf также при помощи rm.

И использовала комнаду git add, которая добавляет изменения из рабочего каталога в раздел проиндексированных файлов,а после commit - команды в системе контроля, которая фиксирует изменения в репозитории.

В самом конце с помощью команды git pull, которая используется для получения изменений из удалённого репозитория и объединения их с локальной веткой.



# Контрольные вопросы для самопроверки

1. Markdown — это легковесный язык разметки, используемый для форматирования текста. Он прост в использовании и легко конвертируется в HTML. Markdown позволяет структурировать текст с помощью заголовков, списков, ссылок, изображений и прочих элементов оформления.
2. В Markdown начертание шрифтов задается следующим образом:
   * Жирный текст — обрамляется двойными звездочками или двойными подчеркиваниями: жирный или жирный.
   * Курсивный текст — одинарными звездочками или подчеркиваниями:

\*курсив\* или \_курсив\_.

* + Жирный курсив — тройными звездочками: \*жирный курсив\*.

1. Списки оформляются так:
   * Ненумерованные списки — с помощью дефиса, звездочки или плюса:

- элемент, \* элемент, + элемент.

* + Нумерованные списки — просто пронумеровать элементы:
    1. элемент, 2. элемент.

1. Изображения и ссылки оформляются следующим образом:
   * Ссылки: [текст ссылки](URL).
   * Изображения: ![текст-заместитель](URL\_изображения).
2. Математические формулы в базовом Markdown не поддерживаются напрямую. Для их добавления часто используют расширение, например, с помощью LaTeX в некоторых Markdown-редакторах:
   * Инлайн-формула: $формула$.
   * Блочная формула: $$формула$$.

# Вывод

В результате выполнения данной лабораторной работы я освоила процедуры оформления отчетов с помощью легковесного языка разметки Markdown.

# Список литературы

1. GDB: The GNU Project Debugger. — URL: https://[www.gnu.org/software/gdb/.](http://www.gnu.org/software/gdb/)
2. GNU Bash Manual. — 2016. — URL:

<https://www.gnu.org/software/bash/manual/>.

1. Midnight Commander Development Center. — 2021. — URL: https://midnight- commander. org/.
2. NASM Assembly Language Tutorials. — 2021. — URL: <https://asmtutor.com/>.
3. Newham C. Learning the bash Shell: Unix Shell Programming. — O’Reilly Media, 2005. — 354 с. — (In a Nutshell). — ISBN 0596009658. — URL:

[http://www.amazon.com/Learningbash-Shell-Programming-](http://www.amazon.com/Learningbash-Shell-Programming-Nutshell/dp/0596009658) [Nutshell/dp/0596009658](http://www.amazon.com/Learningbash-Shell-Programming-Nutshell/dp/0596009658).

1. Robbins A. Bash Pocket Reference. — O’Reilly Media, 2016. — 156 с. — ISBN 978-1491941591.
2. The NASM documentation. — 2021. — URL: <https://www.nasm.us/docs.php>.
3. Zarrelli G. Mastering Bash. — Packt Publishing, 2017. — 502 с. — ISBN 9781784396879.
4. Колдаев В. Д., Лупин С. А. Архитектура ЭВМ. — М. : Форум, 2018.
5. Куляс О. Л., Никитин К. А. Курс программирования на ASSEMBLER. — М. : Солон-Пресс, 2017.
6. Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
7. Расширенный ассемблер: NASM. — 2021. — URL: <https://www.opennet.ru/docs/RUS/nasm/>.
8. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX.

— 2-е изд. — БХВПетербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.

1. Столяров А. Программирование на языке ассемблера NASM для ОС Unix.
   * 2-е изд. — М. : МАКС Пресс, 2011. — URL:

<http://www.stolyarov.info/books/asm_unix>.

1. Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013.
   * 874 с. — (Классика Computer Science).
2. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер, 2015. — 1120 с. — (Классика Computer Science).