Отчёт по лабораторной работе №3

Дисциплина: архитектура компьютеров и операционные системы

Авдадаев Джамал Геланиевич

Содержание

[1 Цель работы 1](#_Toc153762803)

[2 Задание 1](#_Toc153762804)

[3 Теоретическое введение 1](#_Toc153762805)

[4 Выполнение лабораторной работы 2](#_Toc153762806)

[4.1 Обновление локального репозитория 2](#_Toc153762807)

[4.2 Создание отчетов в формате .pdf и .docx 2](#_Toc153762808)

[4.3 Заполнения отчета по лабораторной работе 4](#_Toc153762809)

[4.4 Задание для самостоятельной работы 6](#_Toc153762810)

[5 Выводы 7](#_Toc153762811)

[6 Список литературы 7](#_Toc153762812)

# 1 Цель работы

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

# 2 Задание

1. Обновление локального репозитория.
2. Создание отчетов в формате .pdf и .docx.
3. Заполнения отчета по лабораторной работе.
4. Задание для самостоятельной работы.

# 3 Теоретическое 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. Чтобы создать заголовок, используйте знак #. Чтобы задать для текста полужирное начертание, заключите его в двойные звездочки. Чтобы задать для текста курсивное начертание, заключите его в одинарные звездочки. Чтобы задать для текста полужирное и курсивное начертание, заключите его в тройные звездочки. Блоки цитирования создаются с помощью символа >. 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.

# 4 Выполнение лабораторной работы

## 4.1 Обновление локального репозитория

Открываю терминал, перехожу в каталог курса, сформированный при выполнении лабораторной работы №2 и обновляю локальный репозиторий, скачав изменения из удаленного репозитория с помощью команды git pull. (рис. [10](#fig:001)).

## 4.2 Создание отчетов в формате .pdf и .docx

Перехожу в каталог с шаблоном отчета по лабораторной работе №3 и провожу компиляцию шаблона с использованием Makefile:ввожу команду make и генерирую файлы report.pdf и report.docx. (рис. [10](#fig:001)).

A screenshot of a computer program

Description automatically generatedОткрываю файловый менеджер и проверяю корректность полученных файлов. (рис. [10](#fig:001)).

Проверка создания отчетов

Удалите полученный файлы с помощью команды make clean. (рис. [10](#fig:001)).

Figure 4: Удаление файлов с помощью команды make clean

Проверяю, что файлы report.pdf и report.docx действительно были удалены. (рис. [10](#fig:001)).

A screenshot of a computer

Description automatically generated=

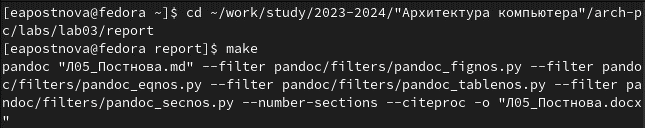
Figure 2: Проверка удаления после создания файлов

## 4.3 Заполнения отчета по лабораторной работе

С помощью текстового редактора открываю файл report.md и внимательно изучаю его структуру, заполняю отчет. (рис. [10](#fig:001)).

Заполнение отчета

Компилирую отчет с использованием Makefile. (рис. [10](#fig:001)).

 Проверяю корректность полученных файлов. (рис. [10](#fig:001)).

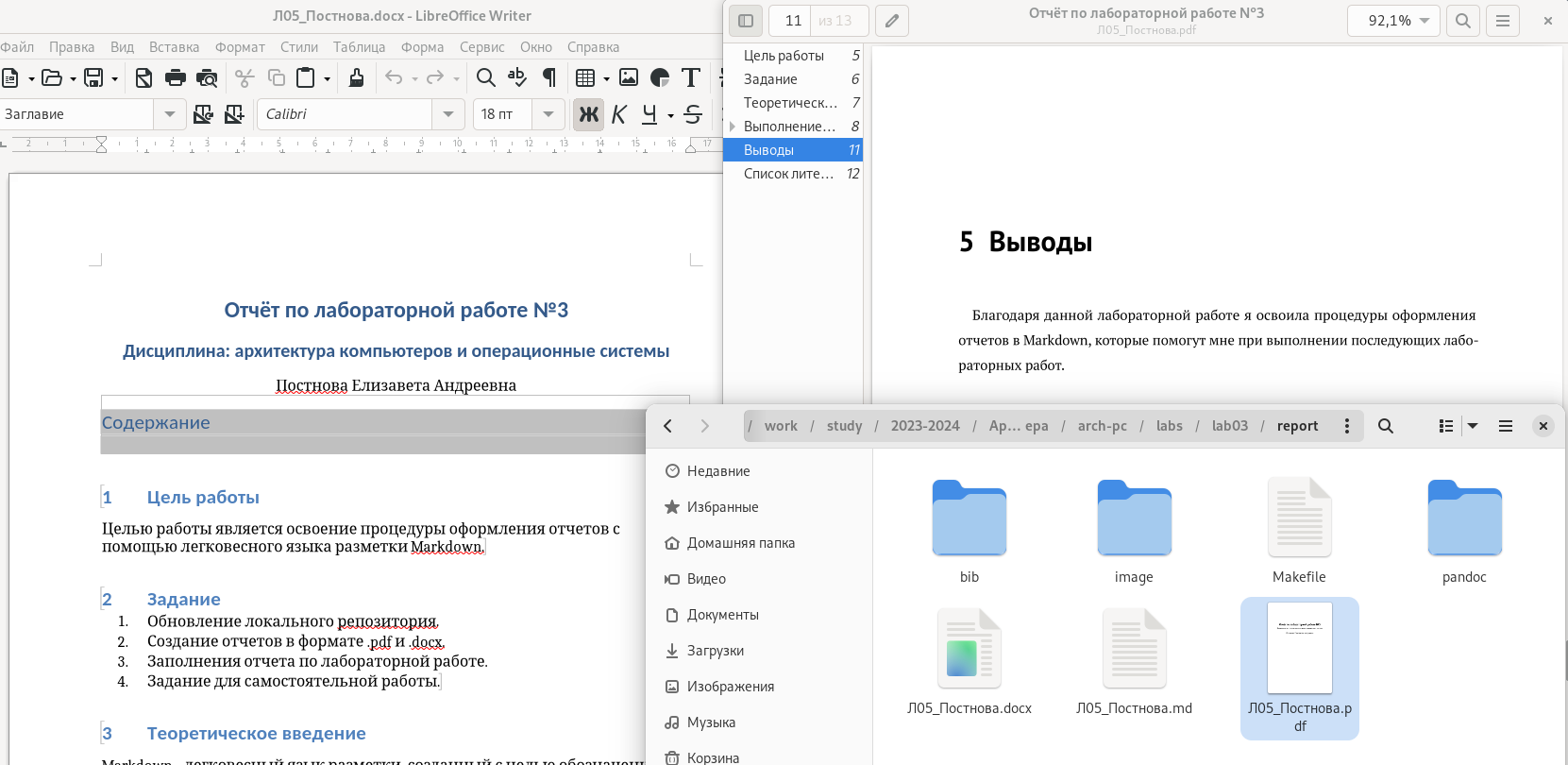


Figure 3: Проверка файлов

Загружаю файлы на Github. (рис. [10](#fig:001)).





Figure 4 Загрузка файлов на Gitgub

## 4.4 Задание для самостоятельной работы

1. В соответствующем каталоге делаю отчёт по лабораторной работе №2 в формате Markdown. (рис. [10](#fig:001)).

A screenshot of a computer

Description automatically generated

Figure 9: Создание отчетов по лабораторной работе №2

1. Загружаю файлы на Github. (рис. [10](#fig:001)).

A screenshot of a computer program

Description automatically generated

Figure 10: Загрузка файлов на Gitgub

# 5 Выводы

Благодаря данной лабораторной работе я освоила процедуры оформления отчетов в Markdown, которые помогут мне при выполнении последующих лабораторных работ В связи с тем что я делал эту лабораторную раньше то некоторые программы не работают.

# 6 Список литературы

1. GDB: The GNU Project Debugger. — URL: https://www.gnu.org/software/gdb/.
2. GNU Bash Manual. — 2016. — URL: https://www.gnu.org/software/bash/manual/.
3. Midnight Commander Development Center. — 2021. — URL: https://midnight-commander. org/.
4. NASM Assembly Language Tutorials. — 2021. — URL: https://asmtutor.com/.
5. 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-Nutshell/dp/0596009658.
6. Robbins A. Bash Pocket Reference. — O’Reilly Media, 2016. — 156 с. — ISBN 978-1491941591.
7. The NASM documentation. — 2021. — URL: https://www.nasm.us/docs.php.
8. Zarrelli G. Mastering Bash. — Packt Publishing, 2017. — 502 с. — ISBN 9781784396879.
9. Колдаев В. Д., Лупин С. А. Архитектура ЭВМ. — М. : Форум, 2018.
10. Куляс О. Л., Никитин К. А. Курс программирования на ASSEMBLER. — М. : Солон-Пресс,
11. Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
12. Расширенный ассемблер: NASM. — 2021. — URL: https://www.opennet.ru/docs/RUS/nasm/.
13. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВПетербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.
14. 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.
15. Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. — 874 с. — (Классика Computer Science).
16. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер 2015. — 1120 с. — (Классика Computer Science).