

# **Отчет по лабораторной работе №3**

**Дисциплина: архитектура компьютеров и операционные системы**

Бизев Никита Владимирович

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
4.1	Обновление локального репозитория . . . . .	8
4.2	Создание отчетов в формате .pdf и .docx . . . . .	9
4.3	Заполнения отчета по лабораторной работе . . . . .	11
<b>5</b>	<b>Выводы</b>	<b>13</b>
	<b>Список литературы</b>	<b>14</b>

## Список иллюстраций

4.1	Использование команды git pull . . . . .	8
4.2	Создание файлов с помощью команды make . . . . .	9
4.3	Проверка создания отчетов . . . . .	10
4.4	Удаление файлов с помощью команды make clean . . . . .	10
4.5	Проверка удаления файлов . . . . .	11
4.6	Заполнение отчета . . . . .	11
4.7	Загрузка файлов на Gitgub . . . . .	12

## **Список таблиц**

# 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.

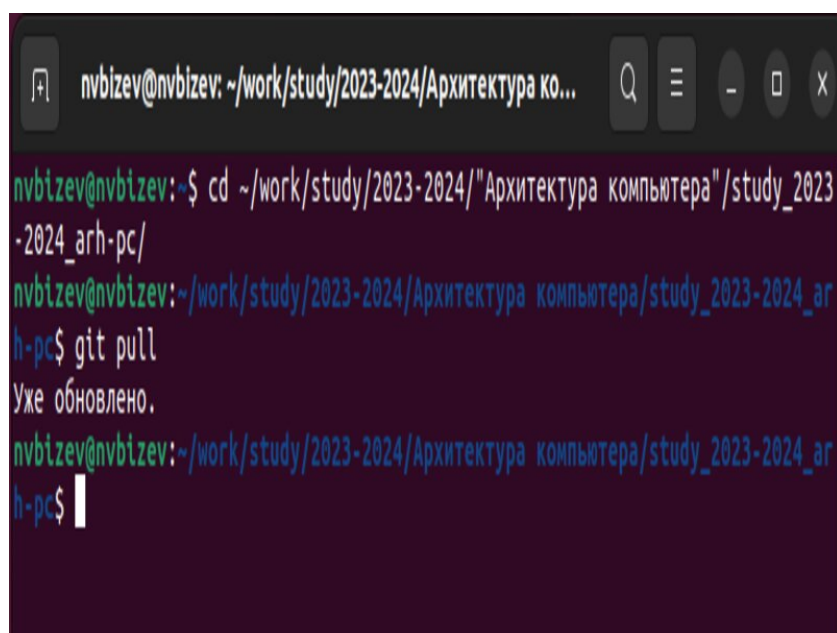
### 3 Теоретическое введение

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. 4.7).

A screenshot of a terminal window with a dark background. The window title bar shows the user 'nvbizev@nvbizev' and the current directory '~/work/study/2023-2024/Архитектура ко...'. The terminal text shows the user navigating to a directory and then running 'git pull'. The output of the command is 'Уже обновлено.' (Already updated).

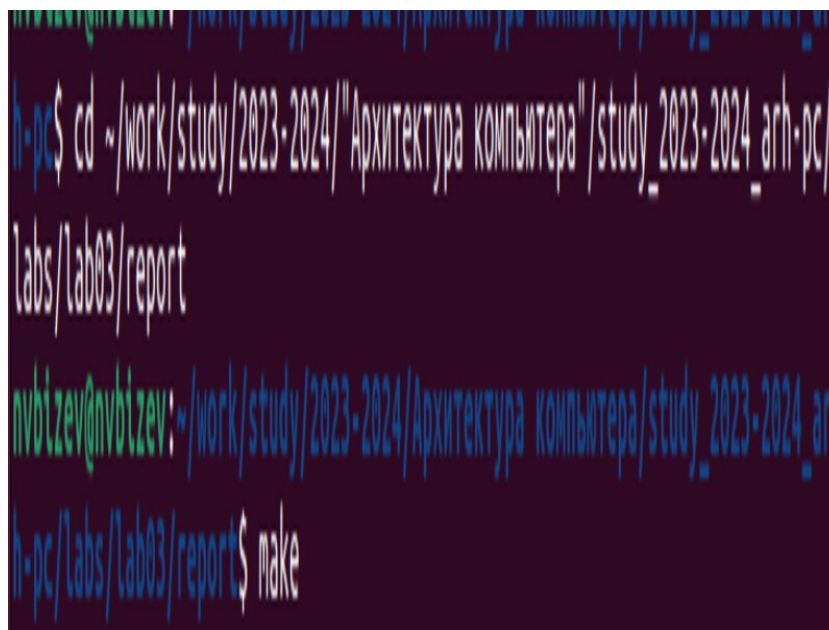
```
nvbizev@nvbizev: ~/work/study/2023-2024/Архитектура ко...
nvbizev@nvbizev:~$ cd ~/work/study/2023-2024/"Архитектура компьютера"/study_2023-2024_arh-pc/
nvbizev@nvbizev:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc$ git pull
Уже обновлено.
nvbizev@nvbizev:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc$
```

Рис. 4.1: Использование команды `git pull`



## 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. 4.7).

A terminal window with a dark background and light-colored text. The prompt is 'h-pc\$'. The user enters 'cd ~/work/study/2023-2024/Архитектура компьютера/study\_2023-2024\_arh-pc/labs/lab03/report'. The prompt changes to 'nvbizev@nvbizev: ~/work/study/2023-2024/Архитектура компьютера/study\_2023-2024\_arh-pc/labs/lab03/report\$'. The user enters 'make'.

```
h-pc$ cd ~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/
labs/lab03/report
nvbizev@nvbizev: ~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_ar
h-pc/labs/lab03/report$ make
```

Рис. 4.2: Создание файлов с помощью команды make

Открываю файловый менеджер и проверяю корректность полученных файлов. (рис. 4.7).

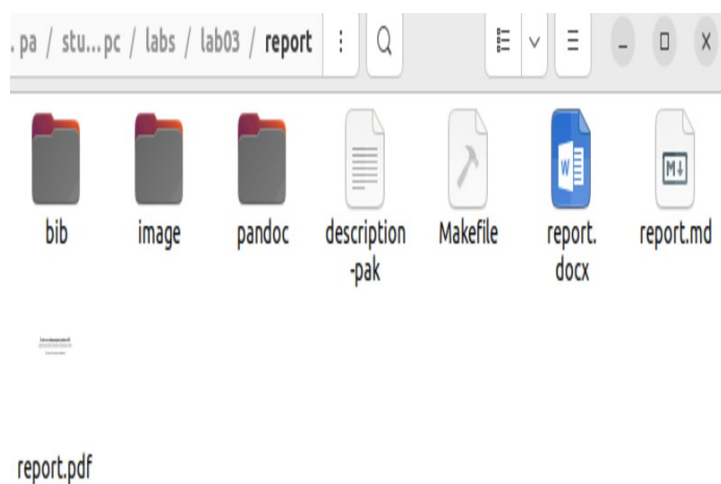


Рис. 4.3: Проверка создания отчетов

Удалите полученные файлы с помощью команды `make clean`. (рис. 4.7).

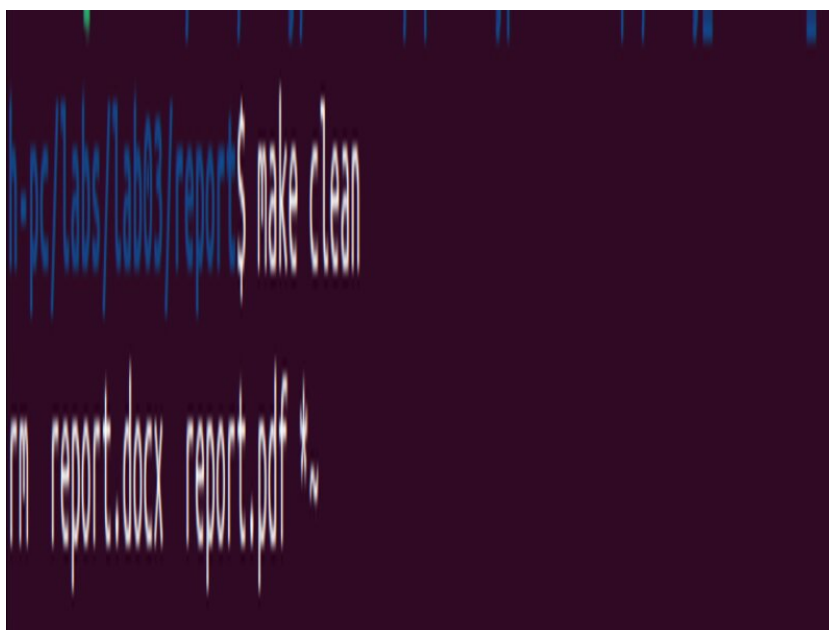


Рис. 4.4: Удаление файлов с помощью команды `make clean`

Проверяю, что файлы `report.pdf` и `report.docx` действительно были удалены. (рис. 4.7).

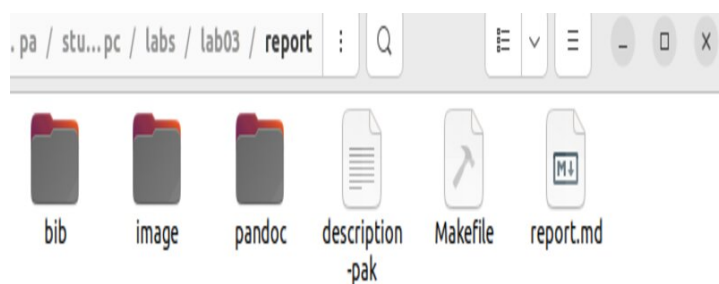


Рис. 4.5: Проверка удаления файлов

### 4.3 Заполнения отчета по лабораторной работе

С помощью текстового редактора открываю файл report.md и внимательно изучаю его структуру, заполняю отчет. (рис. 4.7).

```

101
102 **Создание отчетов в формате .pdf и .docx**
103
104 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. @fig:
    001).
105
106 ![Создание файлов с помощью команды make](image/3.png){#fig:001 width=70%}
107
108 Открываю файловый менеджер и проверяю корректность полученных файлов. (рис. @fig:001).
109
110 ![Проверка создания отчетов](image/4.png){#fig:001 width=60%}
111
112 Удаляю полученные файлы с помощью команды make clean. (рис. @fig:001).
113
114 ![Удаление файлов с помощью команды make clean](image/5.png){#fig:001 width=70%}
115
116 Проверяю, что файлы report.pdf и report.docx действительно были удалены. (рис. @fig:001).
117
118 ![Проверка удаления файлов](image/6.png){#fig:001 width=60%}
119

```

Рис. 4.6: Заполнение отчета

Компилирую отчет с использованием Makefile. (рис. 4.7).

```
h-pc$ cd ~/work/study/2023-2024/"Архитектура компьютера"/study_2023-2024_arh-pc/
labs/lab03/report
nvbizzev@nvbizzev:~/work/study/2023-2024/Архитектура компьютера/study_2023-2024_arh-pc/labs/lab03/report$ make
```

Проверяю корректность

полученных файлов. (рис. 4.7).

Загружаю файлы на Github. (рис. 4.7).

```
nvbizzev@nvbizzev:~$ git add .
nvbizzev@nvbizzev:~$ git commit -m 'feat(main):add files lab-3'
```

Рис. 4.7: Загрузка файлов на Gitgub

## 5 Выводы

Благодаря данной лабораторной работе я освоил процедуры оформления отчетов в Markdown, которые помогут мне при выполнении последующих лабораторных работ.

## 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.
12. Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
13. Расширенный ассемблер: NASM. — 2021. — URL: <https://www.opennet.ru/docs/RUS/nasm/>.
14. Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВПетербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.
15. 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](http://www.stolyarov.info/books/asm_unix).
16. Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. — 874 с. — (Классика Computer Science).
17. Таненбаум Э., Бос Х. Современные операционные системы. — 4-е изд. — СПб. : Питер 2015. — 1120 с. — (Классика Computer Science).