

Отчёт по лабораторной работе №3

Дисциплина: архитектура компьютеров и операционные системы

Хань Цзянтао

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
4.1	Обновление локального репозитория	8
4.2	Создание отчетов в формате .pdf и .docx	8
4.3	Заполнения отчета по лабораторной работе	10
4.4	Задание для самостоятельной работы	11
5	Выводы	13
6	Список литературы	14

Список иллюстраций

4.1	Использование команды git pull	8
4.2	Создание файлов с помощью команды make	8
4.3	Проверка создания отчетов	9
4.4	Удаление файлов с помощью команды make clean	9
4.5	Проверка удаления файлов	9
4.6	Заполнение отчета	10
4.7	Проверка файлов.....	10
4.8	Загрузка файлов на Gitgub	11
4.9	Создание отчетов по лабораторной работе №2	11
4.10	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. 4.10).

```
[eapostnova@fedora ~]$ cd ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/
[eapostnova@fedora arch-pc]$ git pull
Обновление b3a6cfe..6f69e3d
Fast-forward
.../report/Лабораторная работа №1 Постнова Елизавета Андреевна.pdf | Bin 0 -> 977124 bytes
.../report/Лабораторная работа №2 Постнова Елизавета Андреевна.pdf | Bin 0 -> 1562790 bytes
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 labs/lab01/report/Лабораторная работа №1 Постнова Елизавета Андреевна.pdf
create mode 100644 labs/lab02/report/Лабораторная работа №2 Постнова Елизавета Андреевна.pdf
[eapostnova@fedora arch-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. 4.10).

```
[eapostnova@fedora ~]$ cd ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab03/report
[eapostnova@fedora report]$ make
pandoc "report.md" --filter pandoc/filters/pandoc_fignos.py --filter pandoc/filters/pandoc_eqnos.py --filter pandoc/filters/pandoc_tablenos.py --filter pandoc/filters/pandoc_secnos.py --number-sections --citeproc -o "report.docx"
pandoc "report.md" --filter pandoc/filters/pandoc_fignos.py --filter pandoc/filters/pandoc_eqnos.py --filter pandoc/filters/pandoc_tablenos.py --filter pandoc/filters/pandoc_secnos.py --pdf-engine=lualatex --pdf-engine-opt=--shell-escape --citeproc --number-sections -o "report.pdf"
```

Рис. 4.2: Создание файлов с помощью команды `make`

Открываю файловый менеджер и проверяю корректность полученных файлов.
(рис. 4.10).

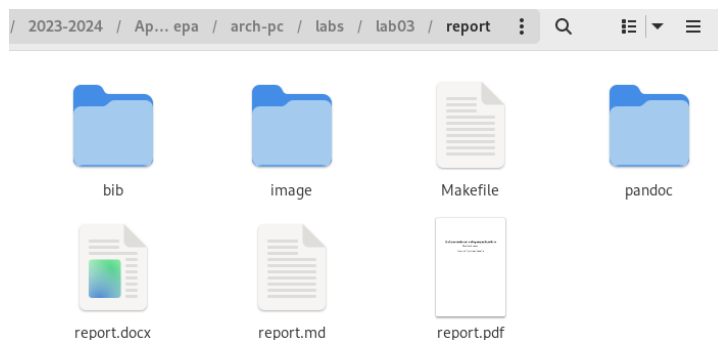


Рис. 4.3: Проверка создания отчетов

Удалите полученные файлы с помощью команды make clean. (рис. 4.10).

```
[earpostnova@fedora report]$ make clean  
rm report.docx report.pdf *~
```

Рис. 4.4: Удаление файлов с помощью команды make clean

Проверяю, что файлы report.pdf и report.docx действительно были удалены.
(рис. 4.10).

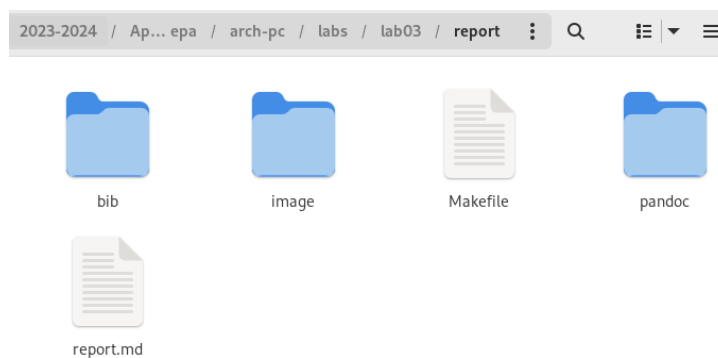
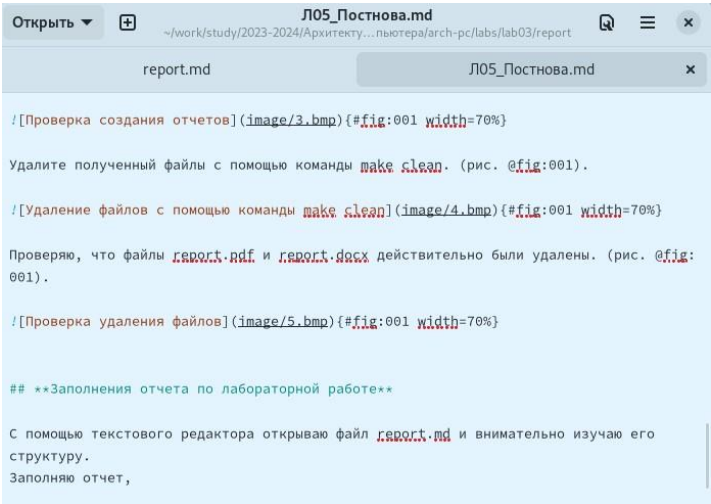


Рис. 4.5: Проверка удаления файлов

4.3 Заполнения отчета по лабораторной работе

С помощью текстового редактора открываю файл report.md и внимательно изучаю его структуру, заполняю отчет. (рис. 4.10).



```
Открыть + Л05_Постнова.md
~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab03/report

report.md Л05_Постнова.md x

/[Проверка создания отчетов](image/3.bmp){#fig:001 width=70%}

Удалите полученный файлы с помощью команды make clean. (рис. @fig:001).

/[Удаление файлов с помощью команды make clean](image/4.bmp){#fig:001 width=70%}

Проверяю, что файлы report.pdf и report.docx действительно были удалены. (рис. @fig:001).

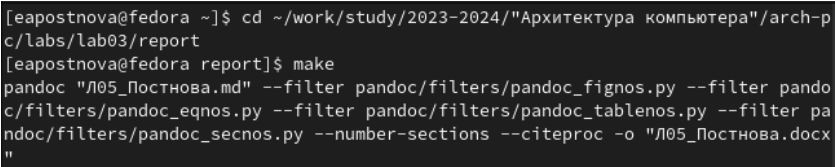
/[Проверка удаления файлов](image/5.bmp){#fig:001 width=70%}

## **Заполнения отчета по лабораторной работе**

С помощью текстового редактора открываю файл report.md и внимательно изучаю его структуру.
Заполняю отчет,
```

Рис. 4.6: Заполнение отчета

Компилирую отчет с использованием Makefile. (рис. 4.10).



```
[eapostnova@fedora ~]$ cd ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab03/report
[eapostnova@fedora report]$ make
pandoc "Л05_Постнова.md" --filter pandoc/filters/pandoc_fignos.py --filter pandoc/filters/pandoc_eqnos.py --filter pandoc/filters/pandoc_tablenos.py --filter pandoc/filters/pandoc_secnos.py --number-sections --citeproc -o "Л05_Постнова.docx"
```

Проверяю коррект-

ность полученных файлов. (рис. 4.10).

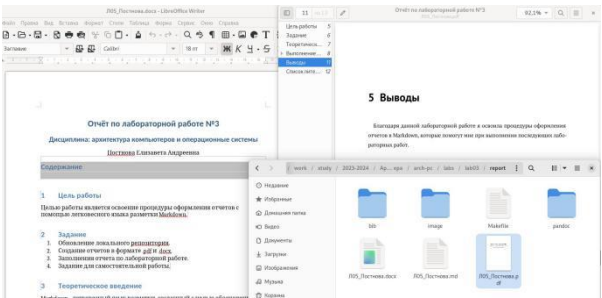


Рис. 4.7: Проверка файлов

Загружаю файлы на Github. (рис. 4.10).

```

[eaipostnova@fedora report]$ cd ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc
[eaipostnova@fedora arch-pc]$ git add .
[eaipostnova@fedora arch-pc]$ git commit -am 'feat(main): add files lab-3'
[master 290199b] feat(main): add files lab-3
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 labs/lab03/report/l05_Постнова.docx
create mode 100644 labs/lab03/report/l05_Постнова.pdf
[eaipostnova@fedora arch-pc]$ git push
Перечисление объектов: 24, готово.
Подсчет объектов: 100% (24/24), готово.
При сжатии изменений используется до 2 потоков
Сжатие объектов: 100% (19/19), готово.
Запись объектов: 100% (19/19), 290.13 КИБ | 2.71 МИБ/с, готово.
Всего 19 (изменений 7), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (7/7), completed with 2 local objects.
To github.com:lisheriz/study_2023-2024_arh-pc.git
6f69e3d..290199b master -> master

```

Рис. 4.8: Загрузка файлов на Github

4.4 Задание для самостоятельной работы

1. В соответствующем каталоге делаю отчёт по лабораторной работе №2 в формате Markdown. (рис. 4.10).

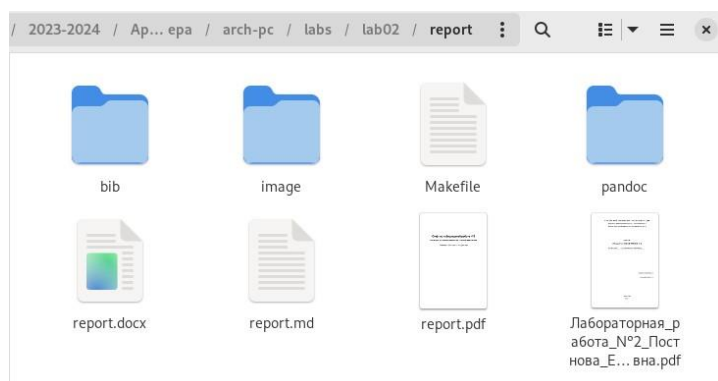


Рис. 4.9: Создание отчетов по лабораторной работе №2

2. Загружаю файлы на Github. (рис. 4.10).

```

[earpostnova@fedora report]$ cd ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc
[earpostnova@fedora arch-pc]$ git add .
[earpostnova@fedora arch-pc]$ git commit -am 'feat(main): add files lab-2'
[master c325e99] feat(main): add files lab-2
35 files changed, 274 insertions(+), 38 deletions(-)
create mode 100644 labs/lab02/report/image/1.png
create mode 100644 labs/lab02/report/image/10.png
create mode 100644 labs/lab02/report/image/11.png
create mode 100644 labs/lab02/report/image/12.png
create mode 100644 labs/lab02/report/image/13.png
create mode 100644 labs/lab02/report/image/14.png
create mode 100644 labs/lab02/report/image/15.png
create mode 100644 labs/lab02/report/image/16.png
create mode 100644 labs/lab02/report/image/17.png
create mode 100644 labs/lab02/report/image/18.png
create mode 100644 labs/lab02/report/image/19.png
create mode 100644 labs/lab02/report/image/2.png
create mode 100644 labs/lab02/report/image/20.png
create mode 100644 labs/lab02/report/image/21.png
create mode 100644 labs/lab02/report/image/22.png
create mode 100644 labs/lab02/report/image/23.png
create mode 100644 labs/lab02/report/image/24.png
create mode 100644 labs/lab02/report/image/3.png
create mode 100644 labs/lab02/report/image/4.png
create mode 100644 labs/lab02/report/image/5.png
create mode 100644 labs/lab02/report/image/6.png
create mode 100644 labs/lab02/report/image/7.png
create mode 100644 labs/lab02/report/image/8.png
create mode 100644 labs/lab02/report/image/9.png
delete mode 100644 labs/lab02/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab02/report/report.docx
create mode 100644 labs/lab02/report/report.pdf
create mode 100644 labs/lab03/report/image/6.bmp
create mode 100644 labs/lab03/report/image/7.bmp
create mode 100644 labs/lab03/report/image/8.bmp
create mode 100644 labs/lab03/report/image/9.bmp
delete mode 100644 labs/lab03/report/Л05_Постнова.docx
delete mode 100644 labs/lab03/report/Л05_Постнова.pdf
[earpostnova@fedora arch-pc]$ git push
Перечисление объектов: 50%, готово.
Подсчет объектов: 100% (50/50), готово.
При сжатии изменений используется до 2 потоков
Сжатие объектов: 100% (41/41), готово.
Запись объектов: 100% (41/41), 3.84 Миб | 3.36 Миб/с, готово.
Всего 41 (изменений 6), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (6/6), completed with 5 local objects.
To github.com:lisheriz/study_2023-2024_arh-pc.git
 290199b..c325e99 master -> master

```

Рис. 4.10: Загрузка файлов на Github

5 Выводы

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

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.
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.
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).