Министерство цифрового развития и массовых коммуникаций

Российской Федерации

Ордена Трудового Красного Знамени федеральное государственное

бюджетное образовательное учреждение

высшего образования

«Московский технический университет связи и информатики»

(МТУСИ)

Кафедра Математическая кибернетика и информационные технологии

**Отчет по лабораторной работе № 1**

по дисциплине «Кроссплатформенные технологии программирования»

на тему: «Основы работы с системой контроля версий»

Выполнила: студентка группы БСТ2105

Мурашова В.А.

Проверила: Смоленцева Т.Е.

Москва 2023

**1. Цель работы**

Изучение базовых функций системы контроля версий на примере Git.

**2. Задание**

Определить проект (состоящий из нескольких файлов), который будет добавлен под версионный контроль, и произвести ряд операций в СКВ, включающих:

1. Создание репозитория или клонирование существующего;
2. Внесение ряда изменений в файл(ы) и выполнение коммитов (кол-во не меньше 2);
3. Создание веток и выполнение ряда изменений в проекте в созданных ветках;
4. Слияние ветки с веткой мастер без конфликтов;
5. Слияние ветки с веткой мастер с решением возникающих конфликтов;
6. Откат изменений до одного из предыдущих коммитов.

**3. Ход работы**

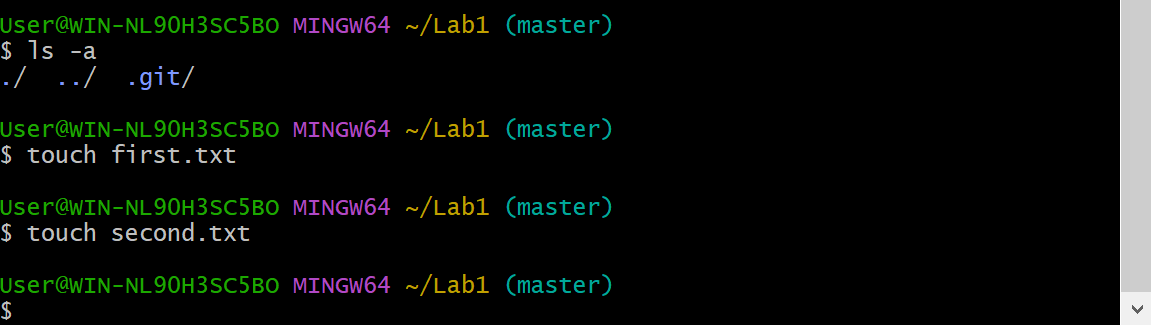
**1. Создание репозитория**

Создаю новый репозиторий в папке Lab1.

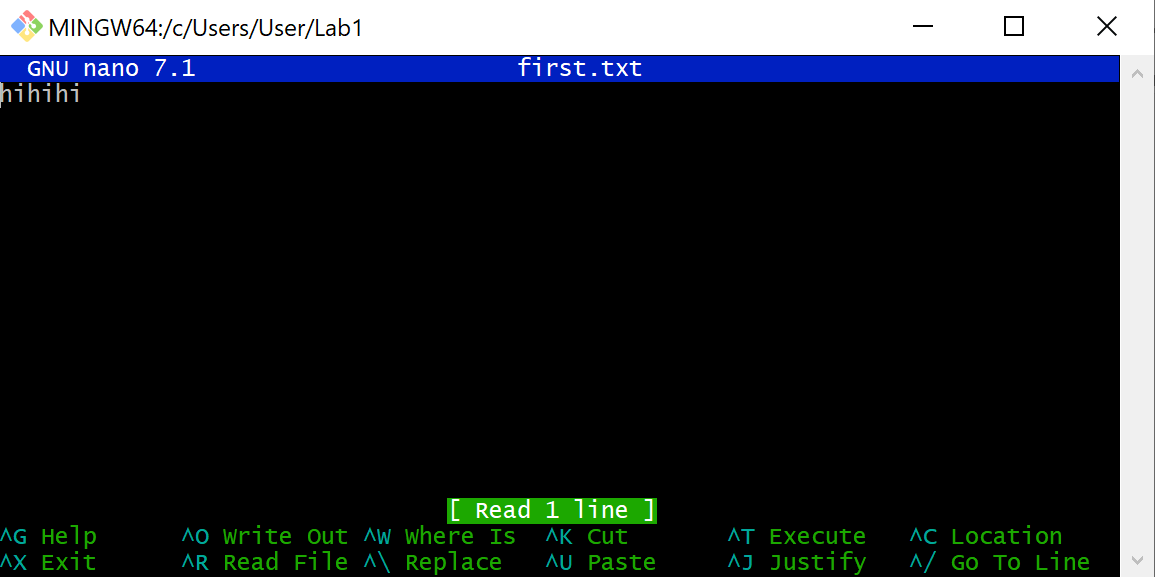


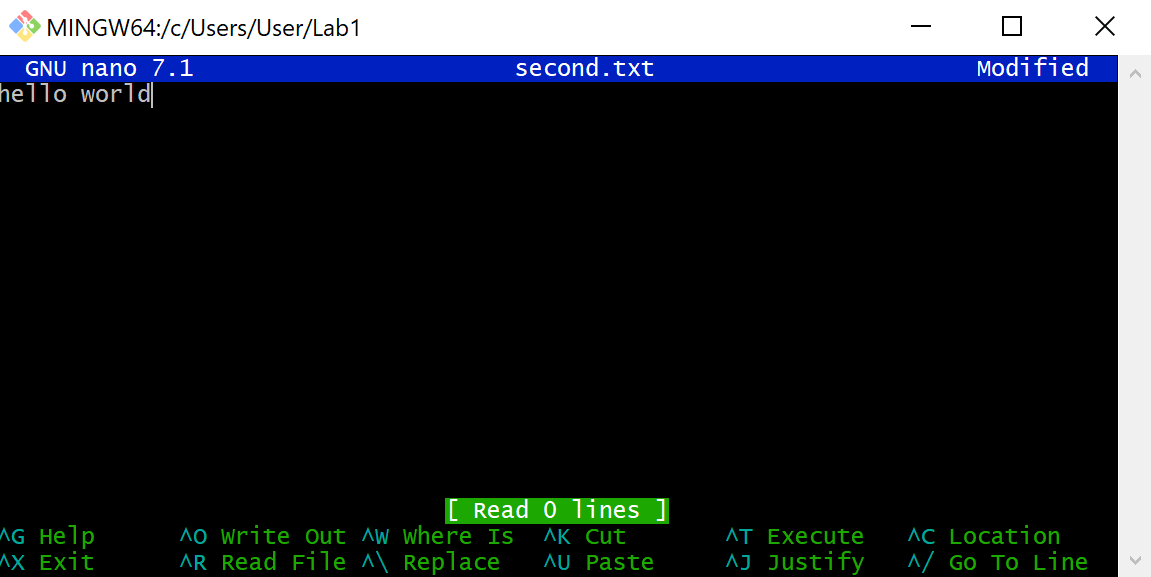
**2. Внесение ряда изменений в файл(ы) и выполнение коммитов**

Проверяю наличие репозитория в папке, создаю два текстовых файла.

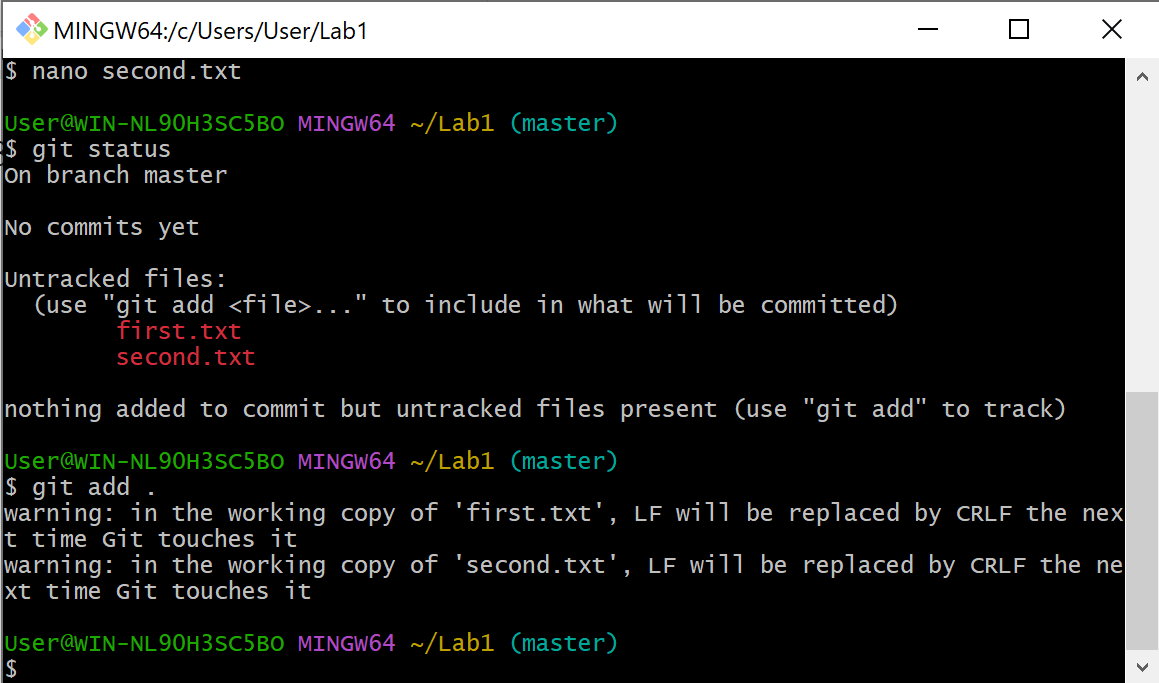


Вношу изменения в текстовые файлы

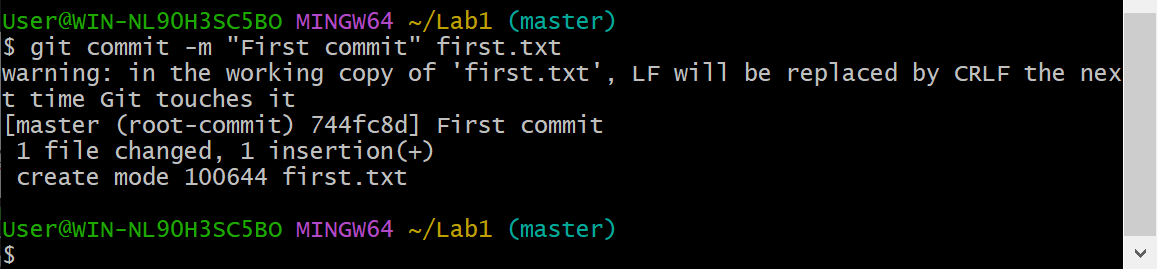




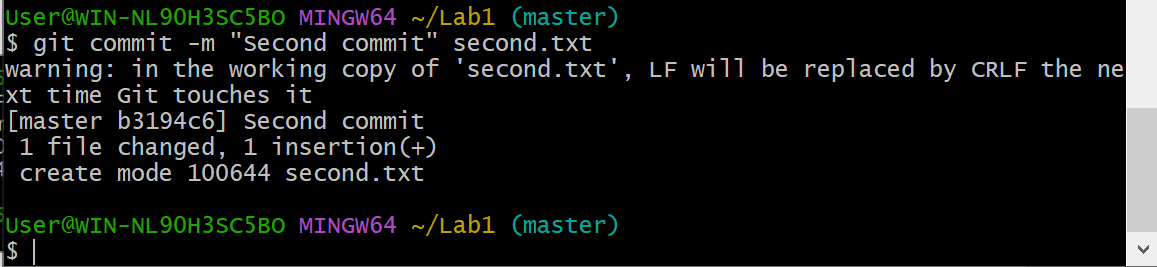
Вношу раннее созданные файлы в контроль версий. (git add .)



Делаю коммит первого файла с комментарием «First commit».

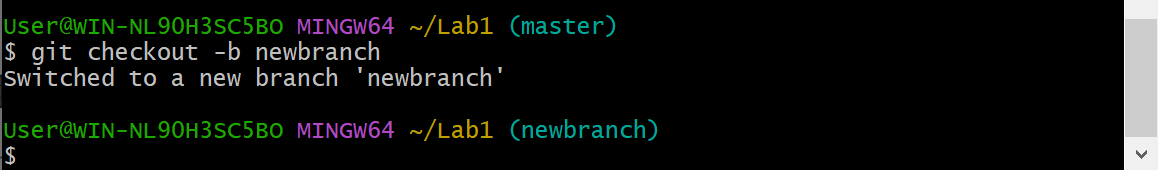


Делаю коммит второго файла с комментарием «Second commit».

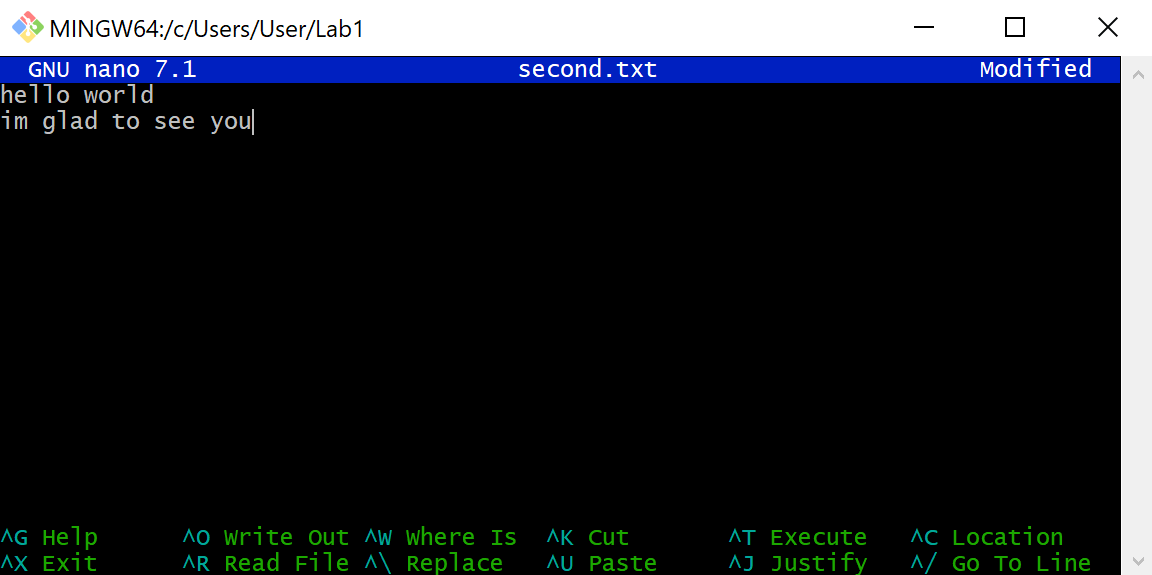


**3. Создание веток и выполнение ряда изменений в проекте в созданных ветках**

Создаю новую ветку и сразу же на нее переключаюсь с помощью команды checkout и флага –b.



Выполняю ряд изменений во втором файле в созданной ветке.

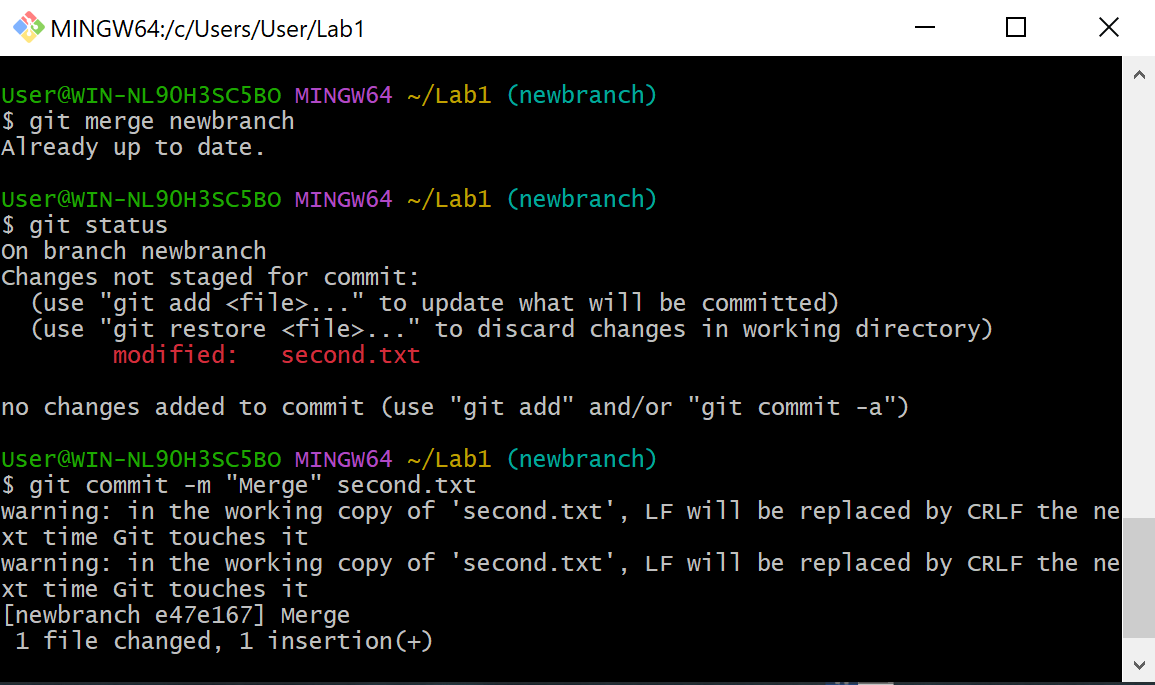


**4. Слияние ветки с веткой мастер без конфликтов**

Находясь в ветке мастер, произвожу слияние с веткой newbranch с помощью команды merge.

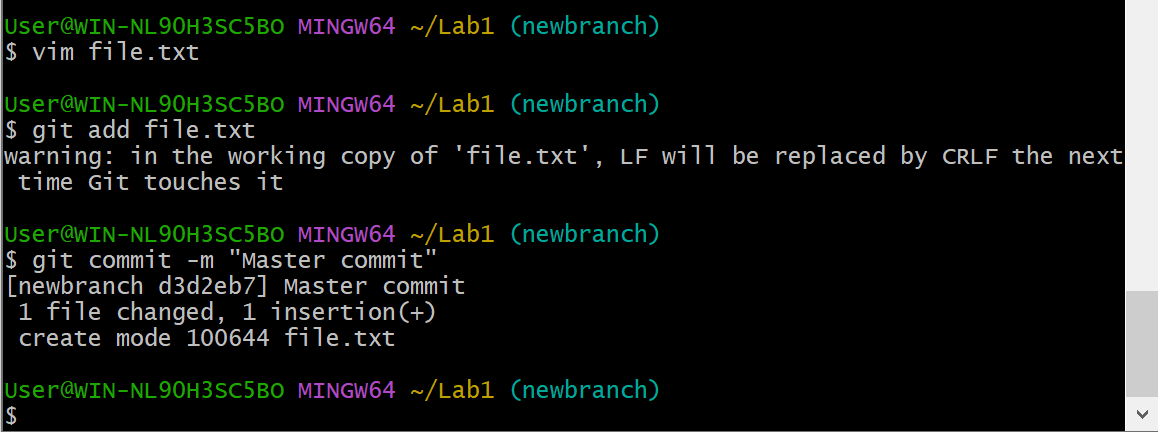
Выполняю коммит первого файла с комментарием «Merge» (после слияния веток).

В результате один измененный файл.

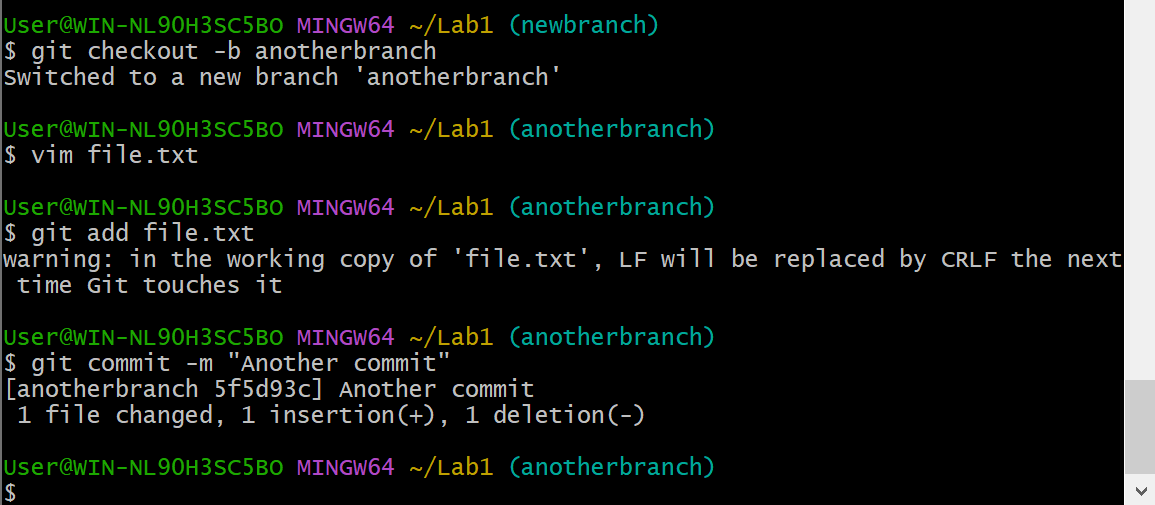


1. **Слияние ветки с веткой мастер с решением возникающих конфликтов**

Для выполнения этого пункта создаю файл file.txt с некоторым содержанием, добавляю и выполняю коммит с комментарием «Master commit».

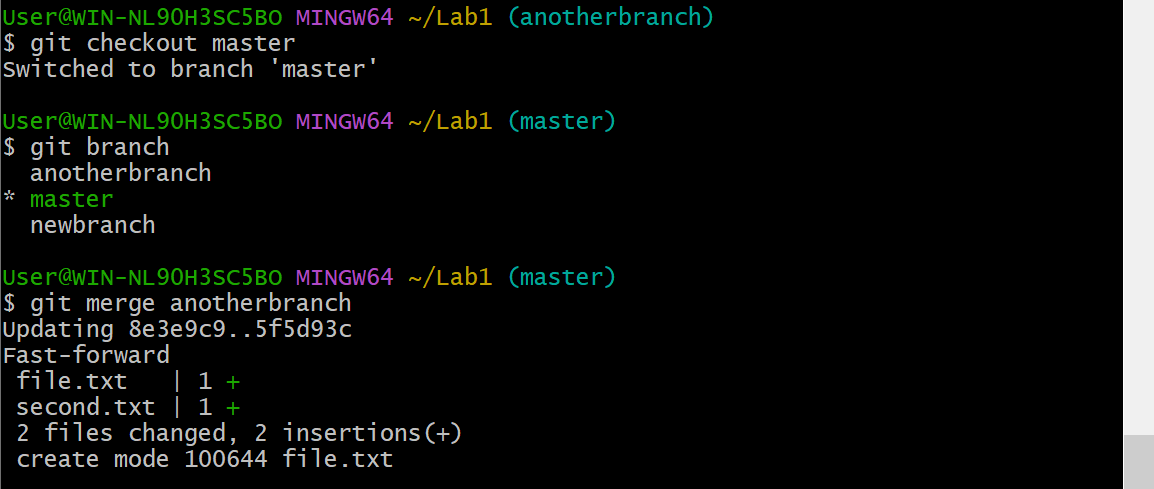


Создаю и переключаюсь на новую ветку anotherbranch. Создаю, редактирую файл file.txt, добавляю и делаю коммит с комментарием “Another commit”.



Переключаюсь на ветку master, выполняю слияние с веткой anotherbranch.

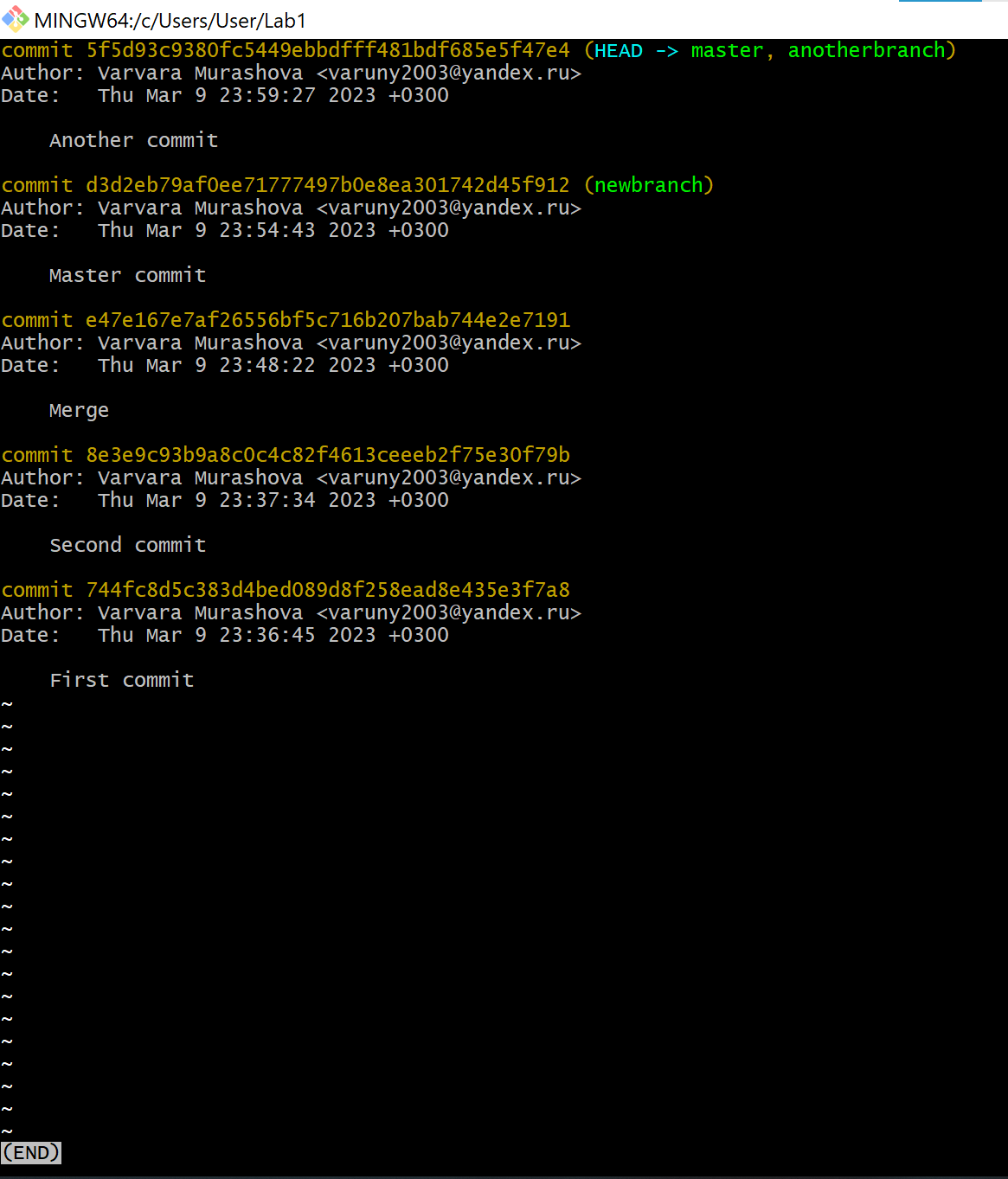
Для решения возникающего конфликта, git автоматически сохранил последний файл с именем file.txt, удалив более поздний.



1. **Откат изменений до одного из предыдущих коммитов**

Смотрим историю версий репозитория с помощью команды log.

Выполняю откат до предыдущего коммита с помощью команды reset в режиме hard.



**4. Вывод**

Я изучила базовые функции системы контроля версий на примере системы контроля версий Git.

Ссылка на репозиторий GitHub: <https://github.com/murrashovaa/Git.git>

**Список используемой литературы**

1. Камаев В.А., Костерин В.В. Технологии программирования. М.: Высшая школа,

2006.

1. Жоголев Е.А.Технология программирования. – М.: Научный мир, 2004.