Лабораторная работа № 1

по дисциплине “Объектно-ориентированное программирование”

Выполнил студент

Группы БИБ1901

Бушеленков Р. С.

Москва 2021

**Основы работы с системой контроля версий**

**Цель работы:** изучение базовых функций системы контроля версий на примере Git.

**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

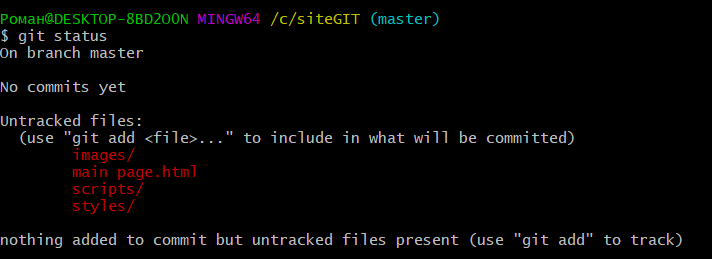
6.Откат изменений до одного из предыдущих коммитов.

**Ход работы:**

1. **Создание репозитория:**



Определение состояния файлов:

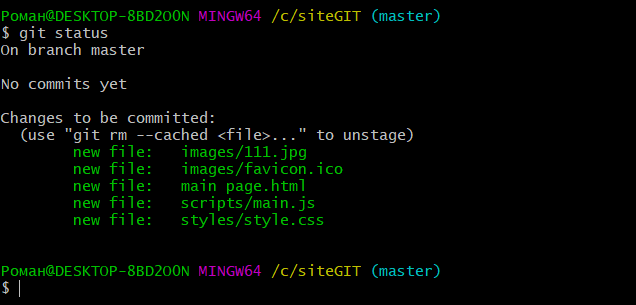


Все файлы проекта не отслеживаются

Добавим все файлы проекта в систему контроля версий:



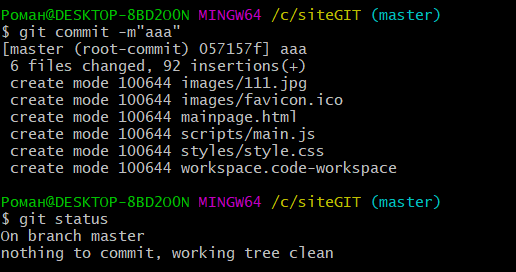
Определение состояния файлов:



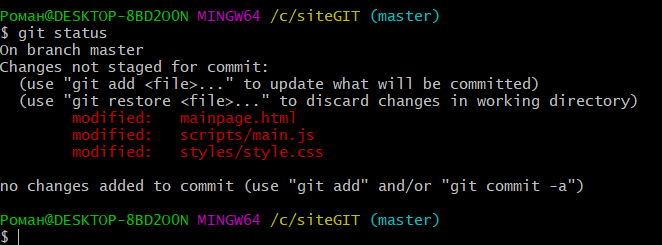
Проект включён в систему контроля версий.

**2.Внесение ряда изменений в файл(ы) и выполнение коммитов (кол-во не меньше 2):**

Сделаем снимок проекта (коммит):



Сделаем изменения в файлах проекта:

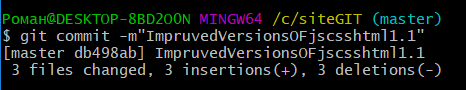


Подготовим изменённые файлы к коммиту:



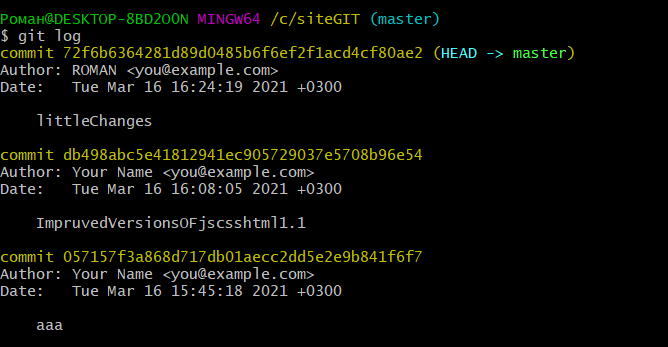
Файлы подготовлены к коммиту.

Выполним коммит:

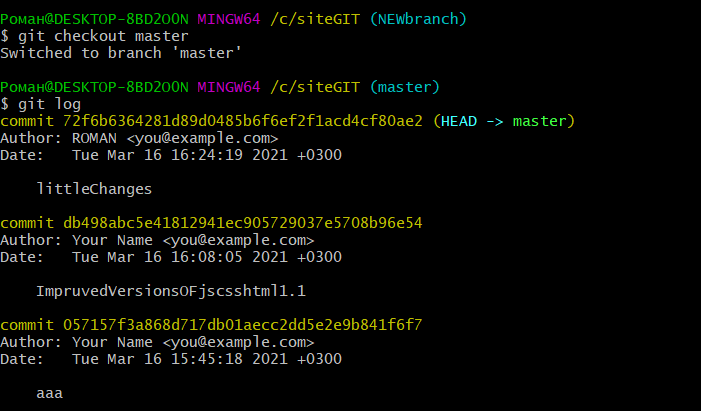
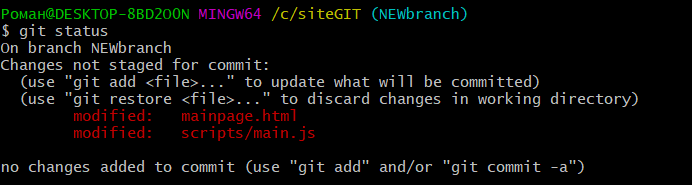
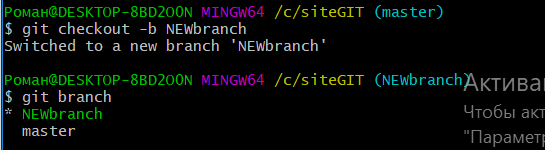
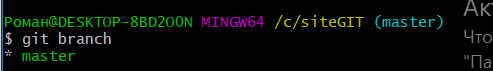


В нашем репозитории теперь 2 коммита. Каждый из них – слепок проекта во времени.

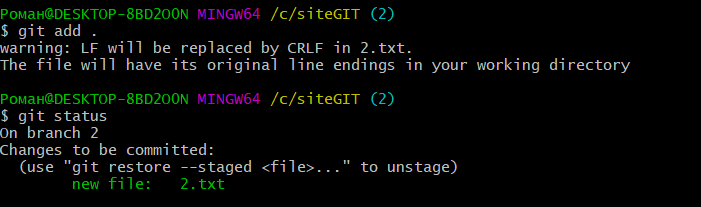
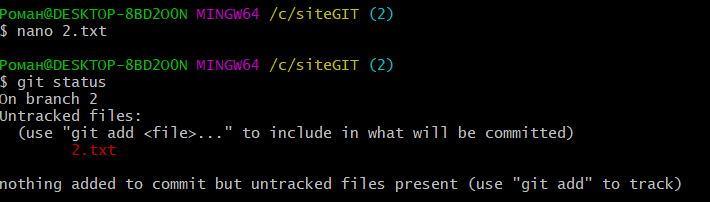
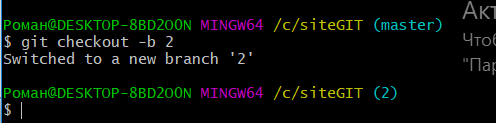
По аналогии выполним третий коммит:



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

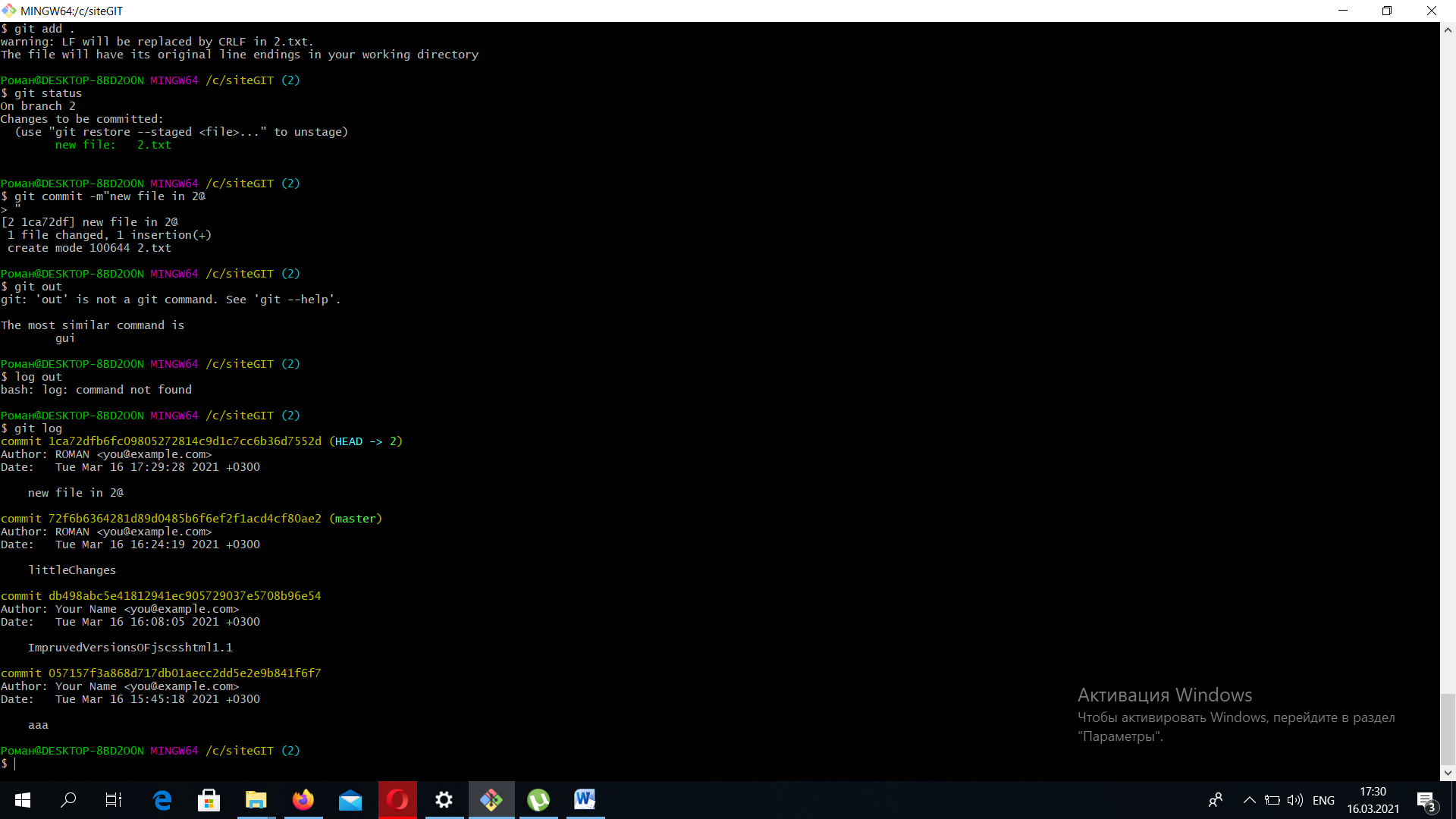


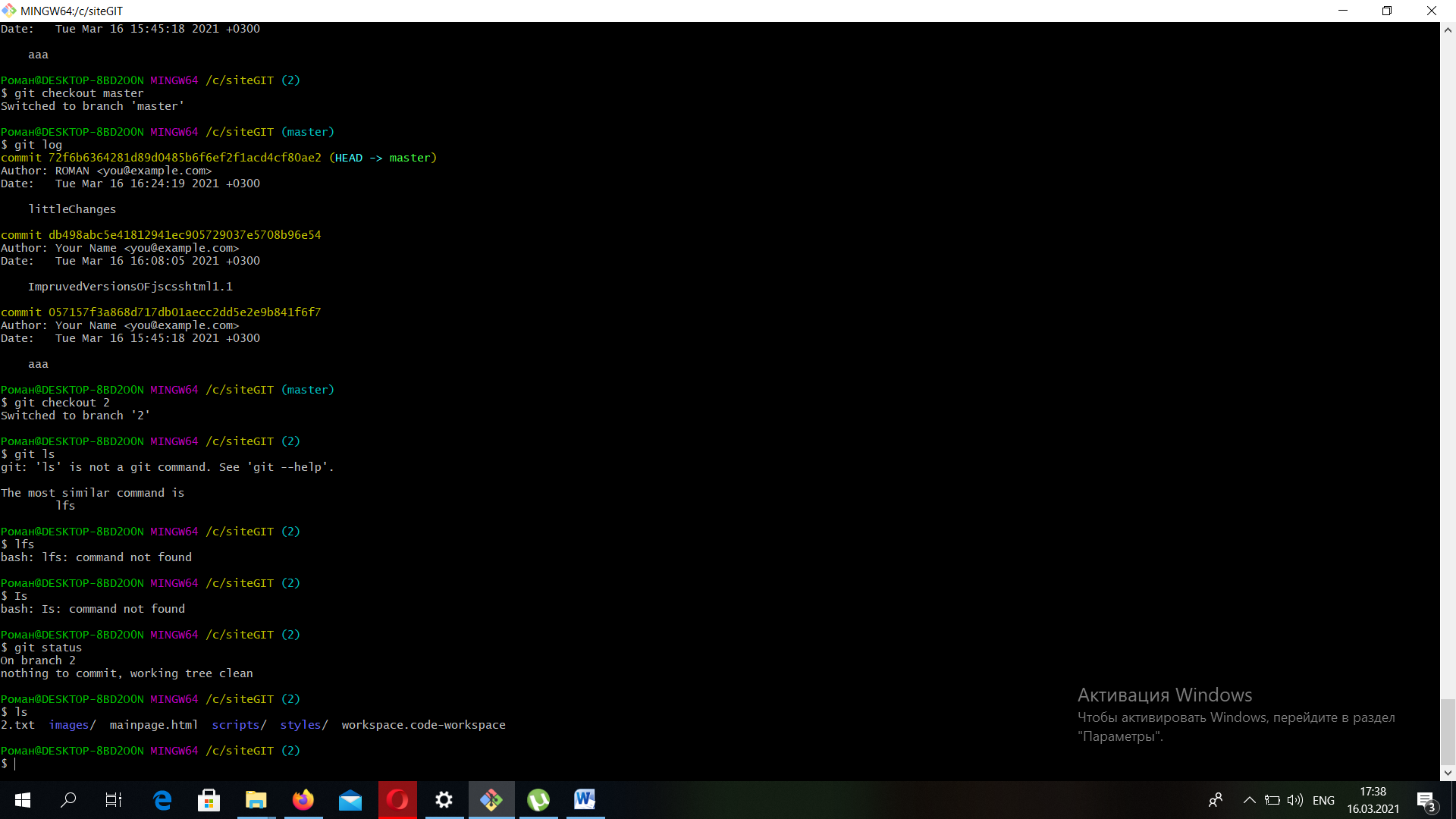
**4.Слияние ветки с веткой мастер без конфликтов**



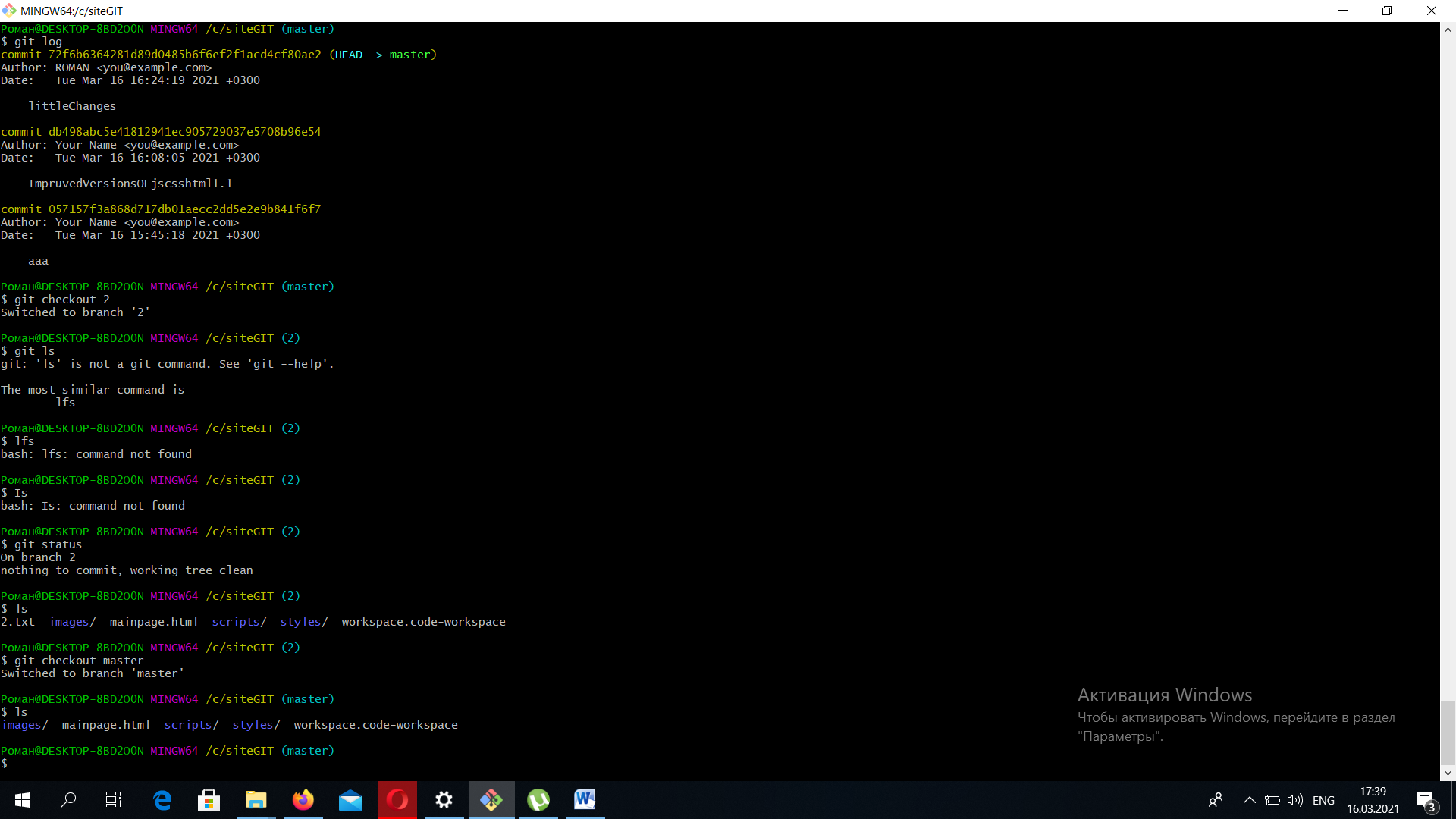
Закоммитим.

Git log:



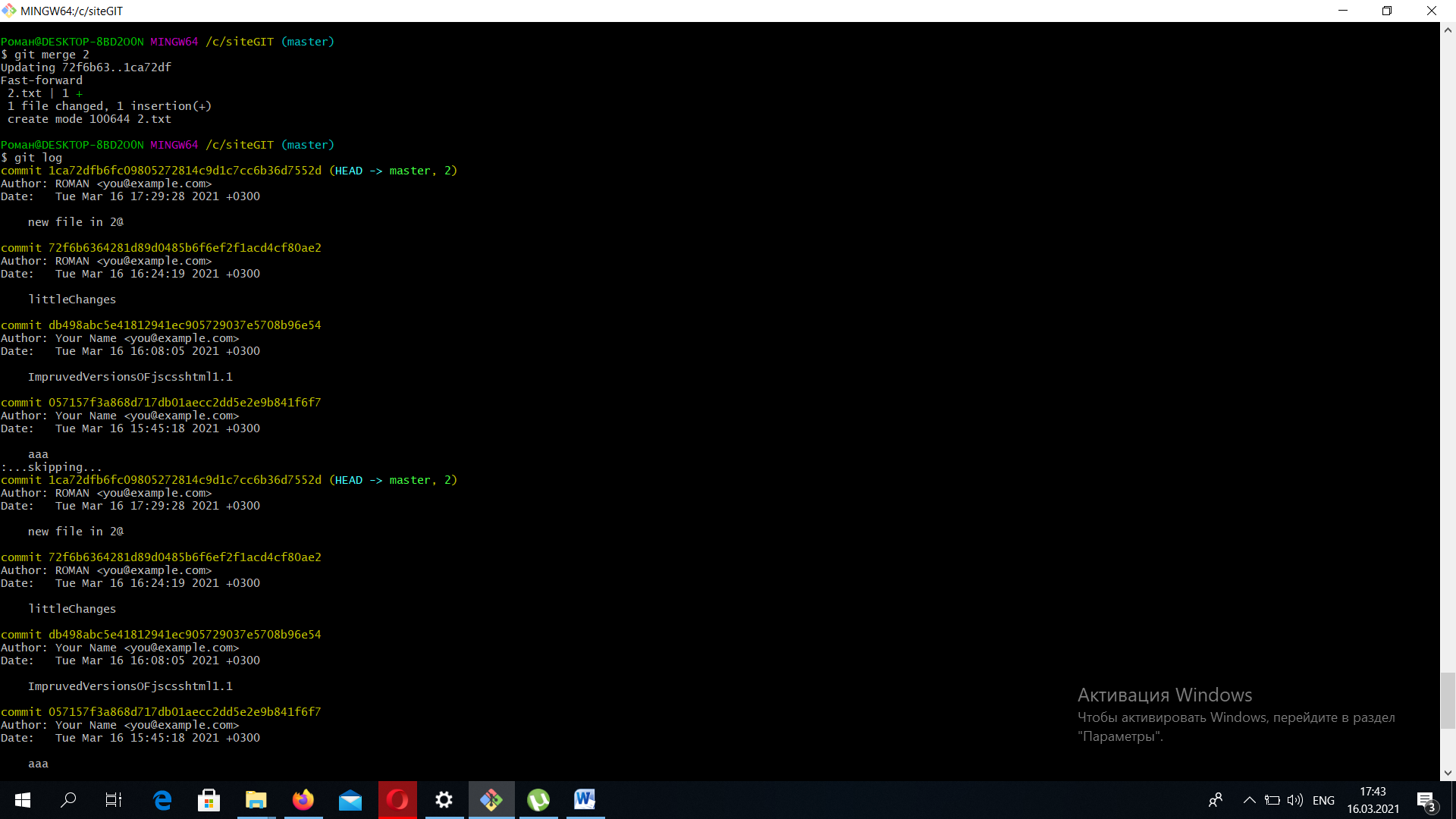


Перейдём в мастер.

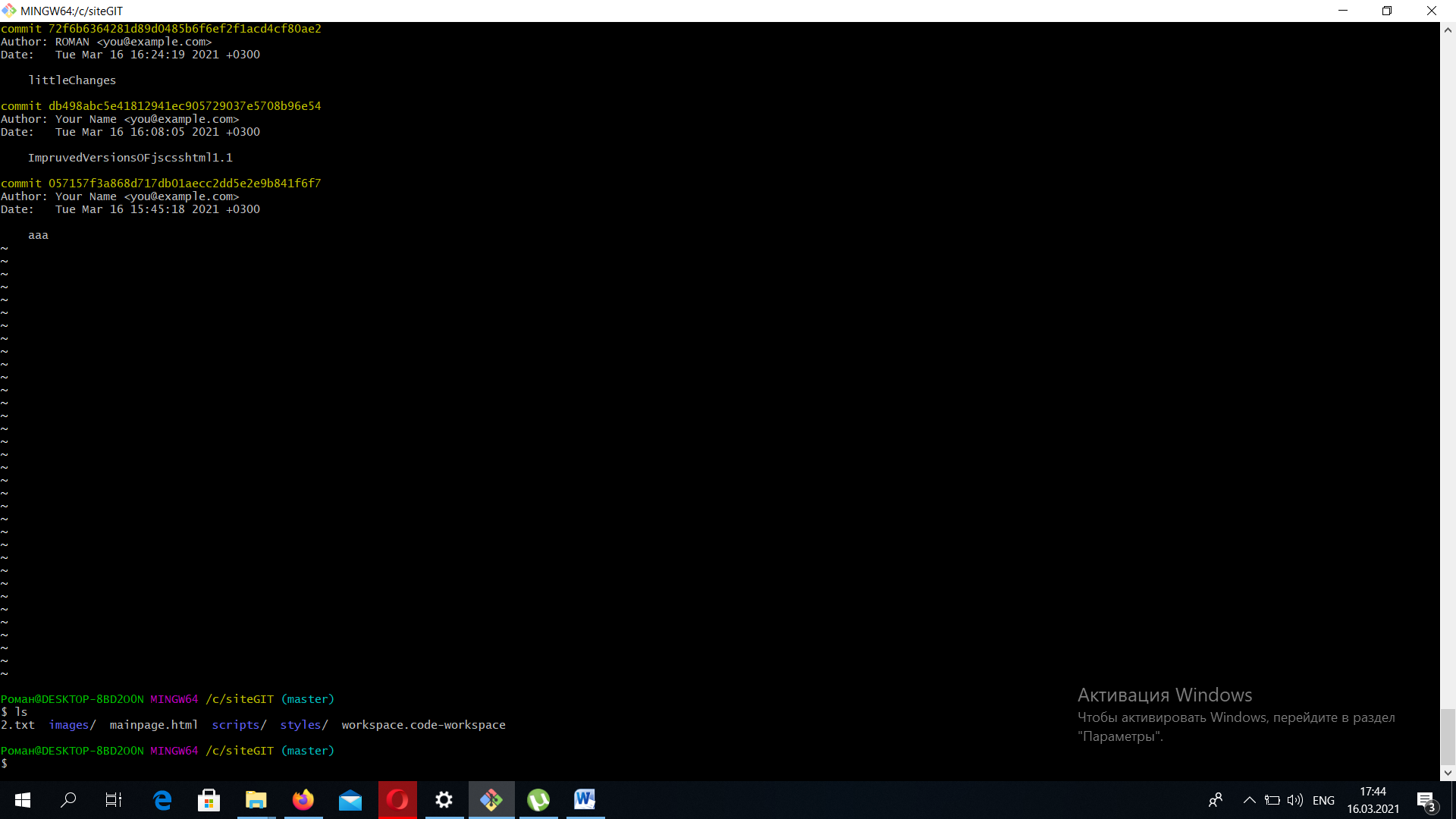


В ветке мастер отсутствует файл 2.txt из ветки 2

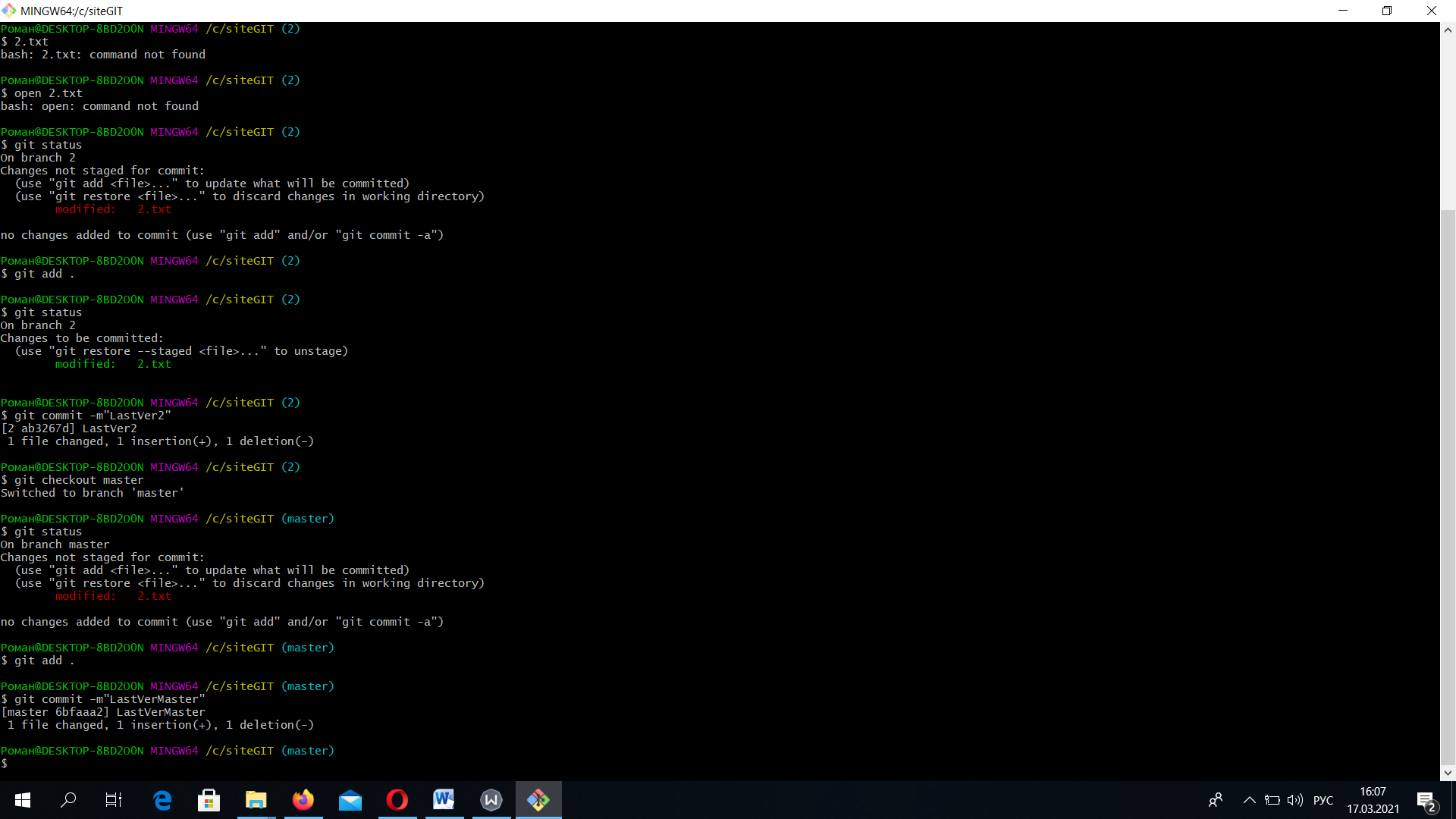
Сольём ветки:



Файл появился в ветке master:

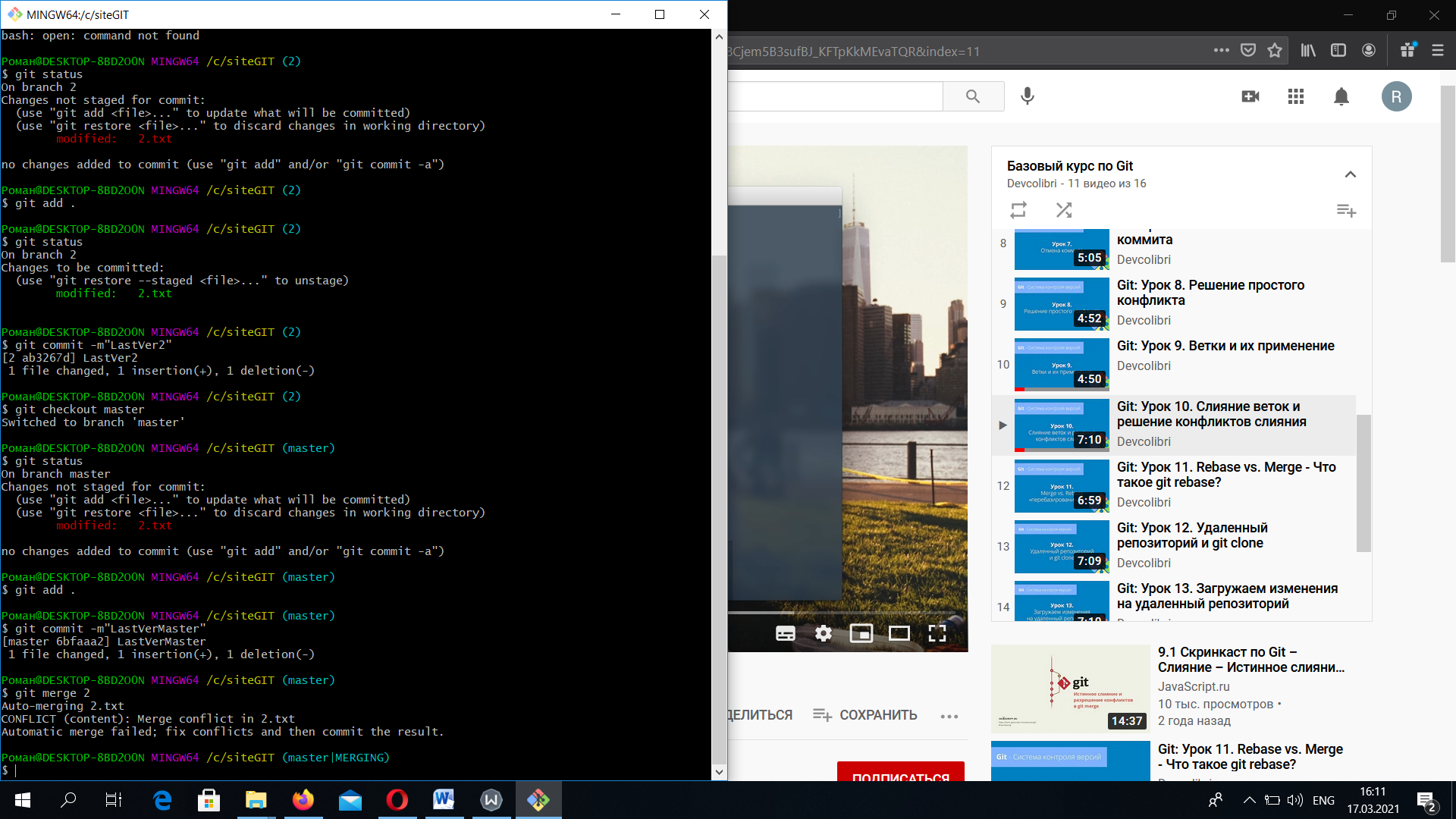


**5.Слияние ветки с веткой мастер с решением возникающих конфликтов.**

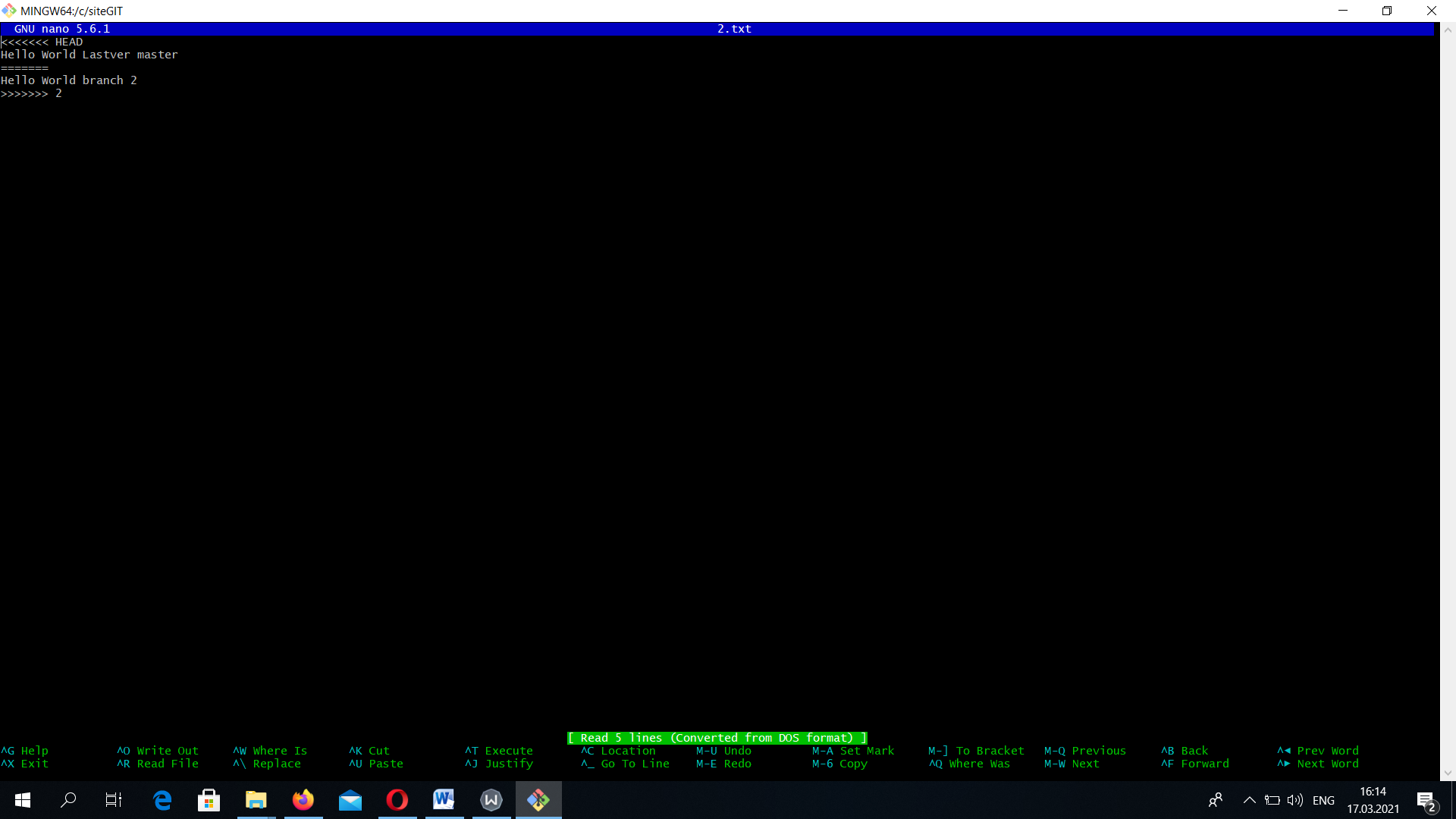


Изменим файл 2.тхт в ветках 2 и master так, чтобы он был различным для каждой ветки.

Попробуем провести слияние веток:

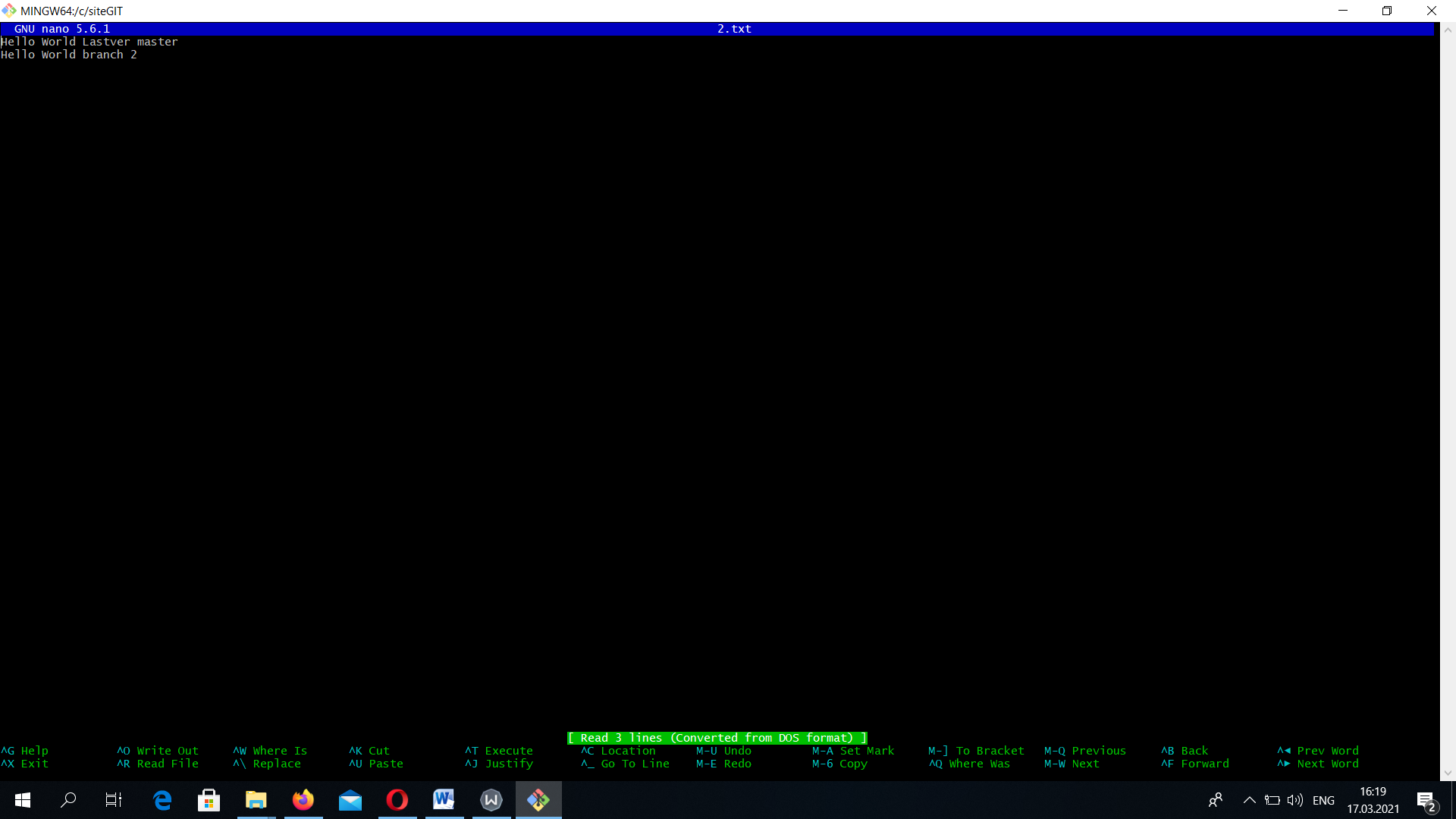


Как и ожидалось, произошёл конфликт при слиянии файлов 2.тхт

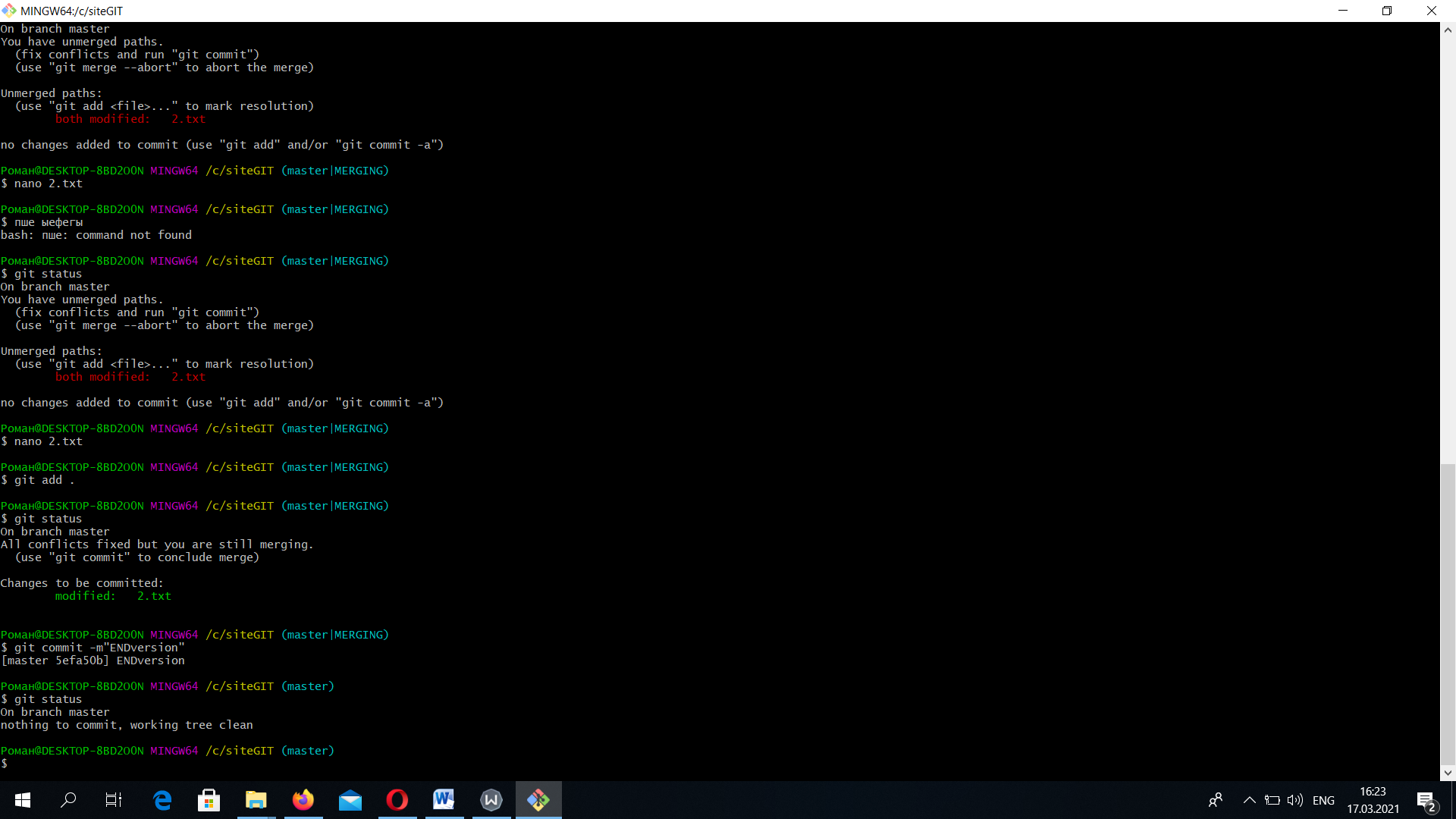


Сверху мы видим то, что было в главной ветке, снизу – что в ветке 2

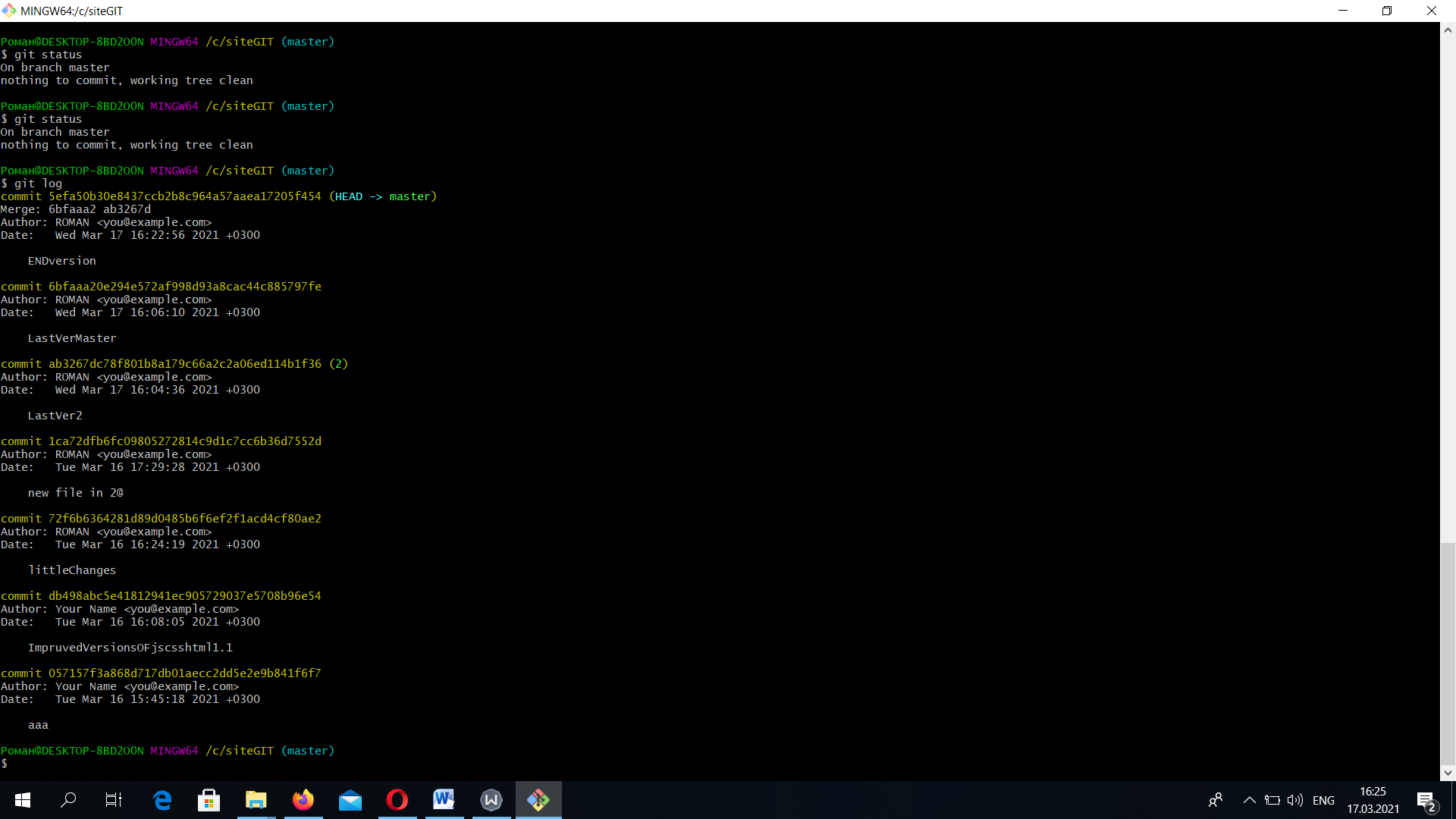
Редактируем файл как хотим, после чего сохраняем



Коммитим файл:



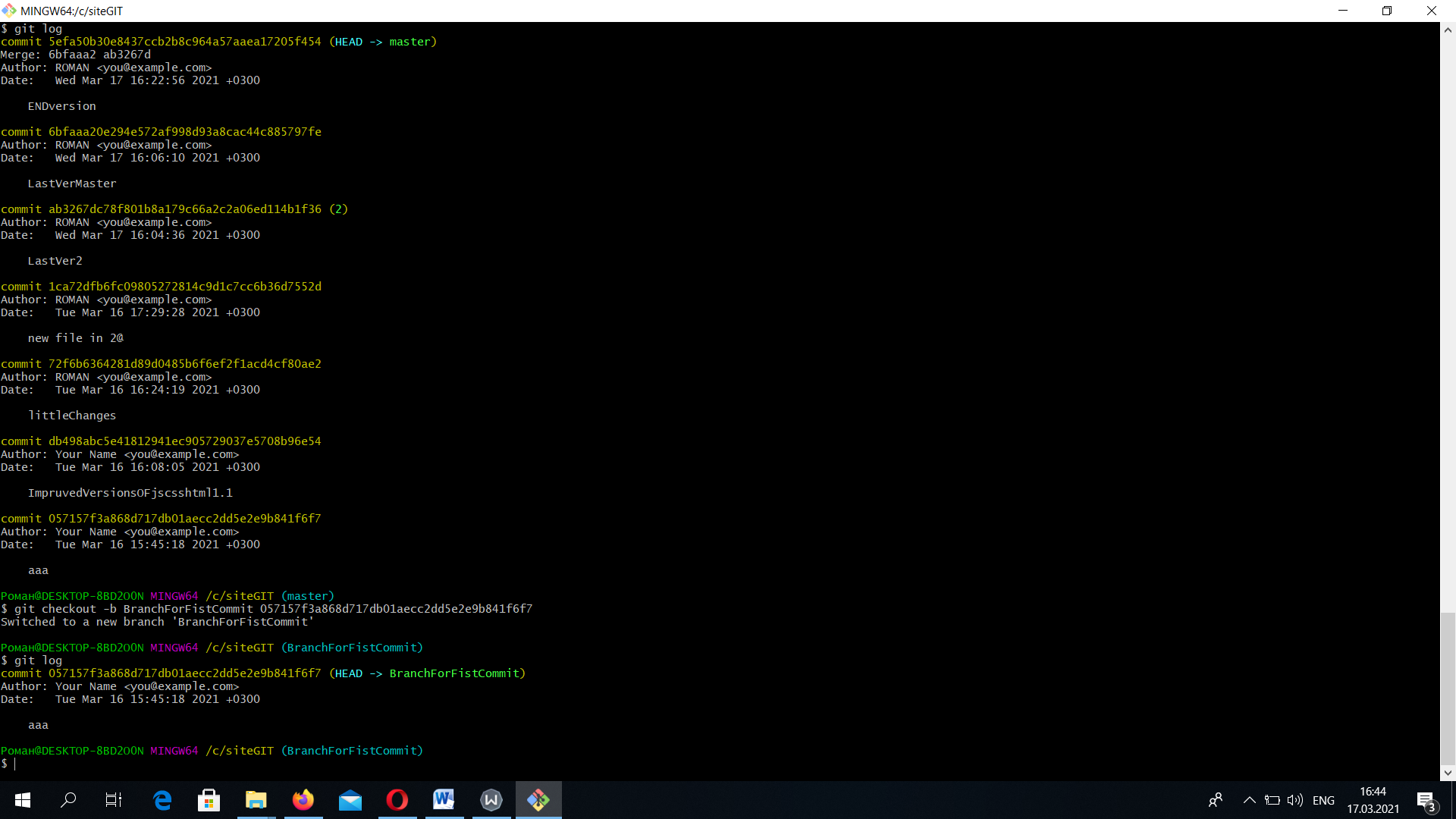
Возникший конфликт успешно решён, ветки слиты:



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

**Чтобы работать с одним из предыдущих коммитов, мы можем создать новую ветку. Можно переключиться и создать ее одной командой:**

**git checkout -b имя-новой-ветки имя-коммита**



Список использованных источников

1) Документация Git: https://git-scm.com/doc

2) Документация GitLab: https://docs.gitlab.com/

3) Java. Полное руководство. Герберт Шилдт