ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ **БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ** (НИУ «БелГУ»)

ИНСТИТУТ ИНЖЕНЕРНЫХ И ЦИФРОВЫХ ТЕХНОЛОГИЙ

КАФЕДРА МАТЕМАТИЧЕСКОГО И ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ

**Отчет по лабораторной работе №5**

**по дисциплине: «Введение в специальность»**

**Тема работы «Ветвление. Конфликты»**

студента очного отделения 2 курса 12001801 группы Данильченко Арины Романовны

Проверил:

асс.Гайворонский Виталий Александрович

Белгород 2019

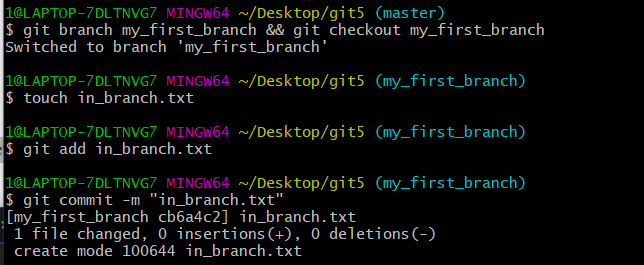
**Лабораторная работа №5**

**Цель работы:** научиться создавать ветки, перемещаться по ним, объединять и удалять их. Решать конфликты слияния.

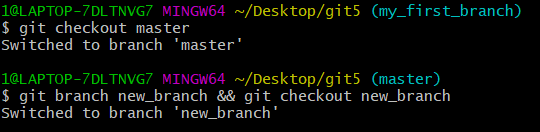
**Выполнение работы**

Выполнила первоначальную настройку репозитория Git и создадала файлы 1.txt, 2.txt и 3.txt.

Создала новую ветку my\_first\_branch с помощью команды git branch. Перешла на неё, создала файл in\_branch.txt и закоммитила изменения. Для перехода на другую ветку использовала команду git checkout:

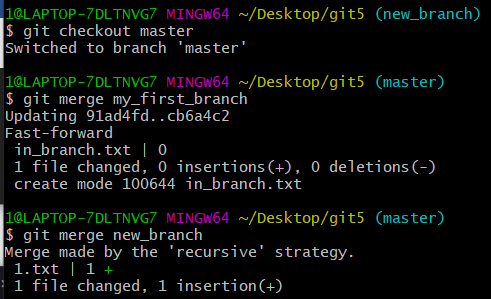


Вернулась на ветку master, создала новую ветку new\_branch и перешла на неё:

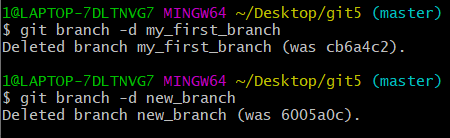


Сделала изменения в файле 1.txt, добавив строку “new row in 1.txt file” и закоммитила изменения.

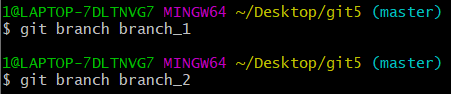
Перешла на ветку master и выполнила слияние веток master и my\_first\_branch. Для слияния использовала команду git merge. Выполнила слияние ветки master и new\_branch:



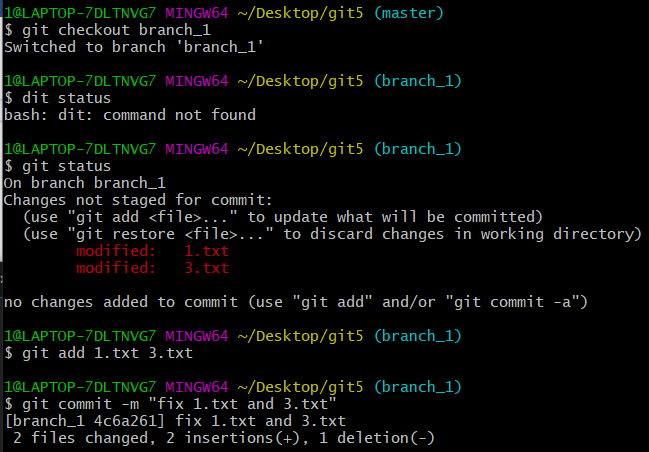
Удалила ветки my\_first\_branch и new\_branch. Удаление веток выполняла с помощью команды git branch с параметром -d:



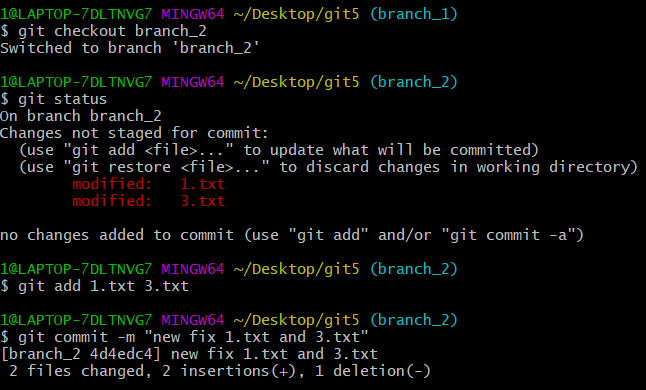
Создала ветки branch\_1 и branch\_2:



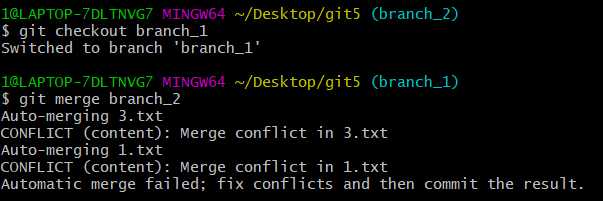
Перешла на ветку branch\_1 и изменила файл 1.txt, удалив все содержимое и добавив текст “fix in 1.txt”, изменила файл 3.txt, удалив все содержимое и добавив текст “fix in 3.txt”, закоммитила изменения:



Перешла на ветку branch\_2 и также изменила файл 1.txt, удалив все содержимое и добавив текст “My fix in 1.txt”, изменила файл 3.txt, удалив все содержимое и добавив текст “My fix in 3.txt”, закоммитила изменения:

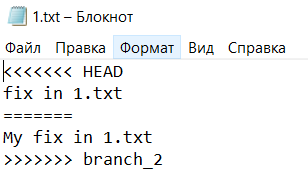


Попыталась произвести слияние изменений ветки branch\_2 в ветку branch\_1. Поступило сообщение о конфликте слияния:

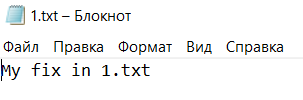


Нужно решить конфликт файла 1.txt в ручном режиме, а конфликт файла 3.txt - используя команду git mergetool

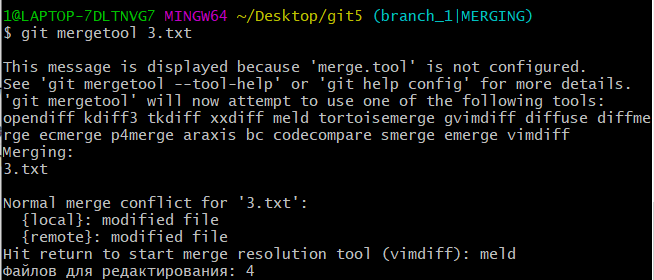
Начала с файла 1.txt. Открыла его в текстовом редакторе:



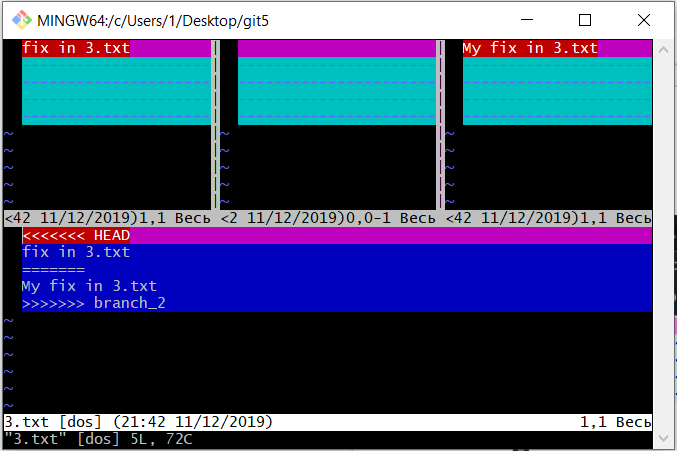
Здесь мы видим 2 версии, которые разделены символами “=”. Нужно выбрать одну из версий, при этом удалив лишнее. Выбирала изменения в ветке branch\_2, т.к. данное изменение является окончательным для данного файла:



На очереди файл 3.txt. Для его изменения использовала специальную команду git mergetool:



После открытия файла через данный инструмент, высветилось следующее окно:



Данные окна интерпретируются следующим образом:

|  |  |  |
| --- | --- | --- |
| Local | Base | Remote |
| Merger | | |

Это 4 вида:

- LOCAL - это файл из текущей ветки;

- BASE - общий предок, как файл выглядел до обоих изменений;

- REMOTE - файл, который объединяете в свою ветку;

- MERGED - результат слияния, это то, что сохраняется в репозитории.

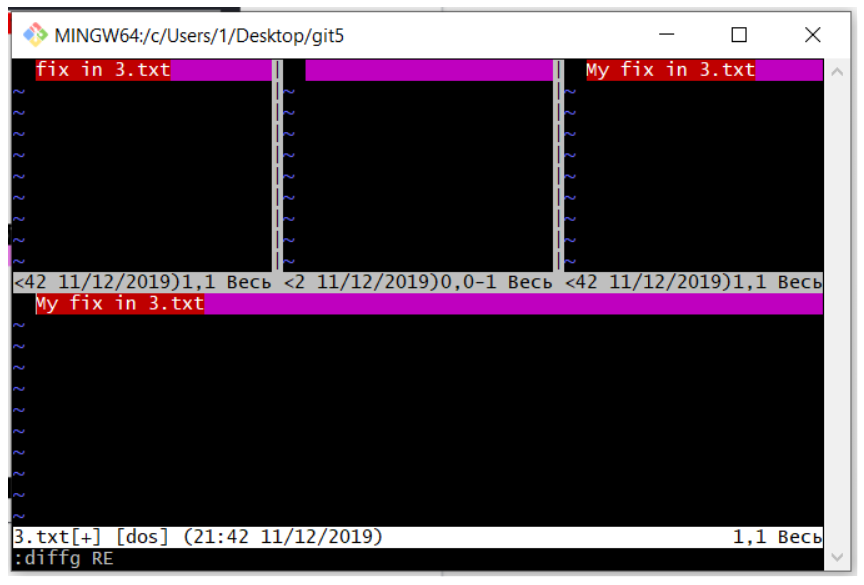
Для работы доступны следующие команды:

- если нужно получить изменения от REMOTE - : diffg RE

- если внужно получить изменения от BASE - : diffg BA

- если нужно получить изменения от LOCAL - :diffg LO

Нам нужны изменения из REMOTE, значить нужно ввести команду :diffg RE:



Вышла с помощью команды :wqa.

Закоммитила данные изменения файлов и выполнила слияние:

