Лабораторная работа №7

Скачиваю Git. Все настройки в инсталляторе оставляю по умолчанию (Рисунок 1).

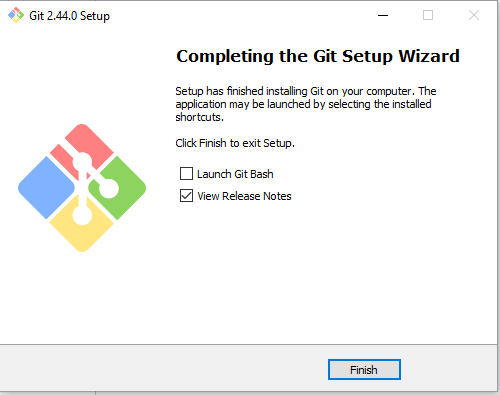


Рисунок 1-Установка Git

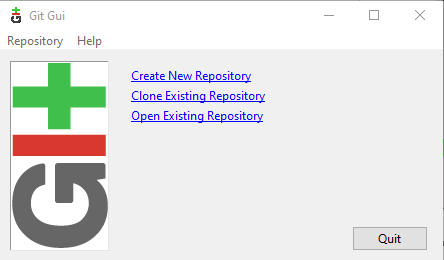


Рисунок 2-Create New Repository

Выбираю Create New Repository (Рисунок 2).

Заполняю путь к моей новой директории и щелкаю Create (Рисунок 3).

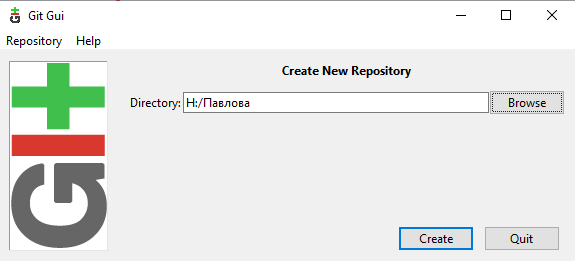


Рисунок 3-Путь к директории

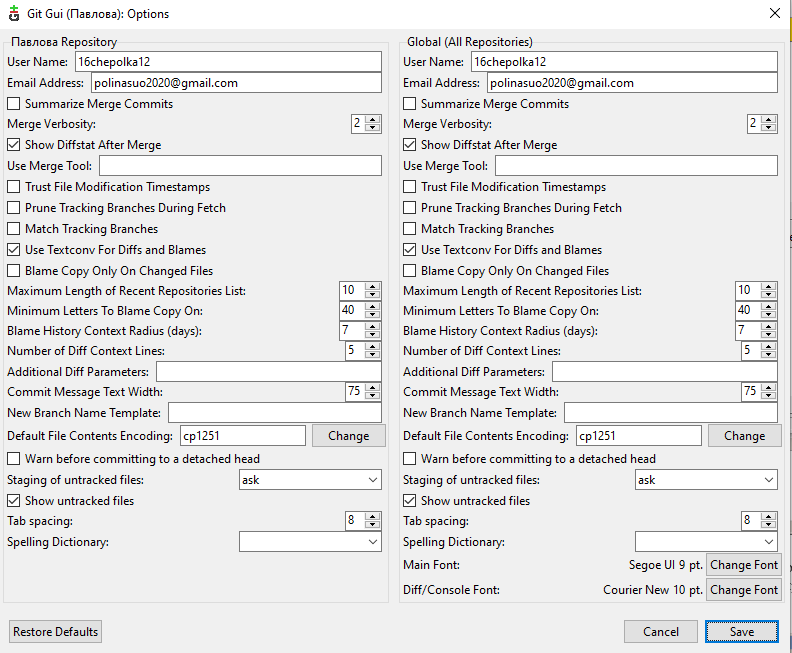
 Выбираю Edit -> Options (Рисунок 4).

Рисунок 4-Настройки

Теперь, когда хранилище создано, создаю файл main.c со следующим содержимым (редактирование выполняю в блокноте). Щёлкаю на кнопку Rescan, git gui нашёл новый файл (Рисунок 5).

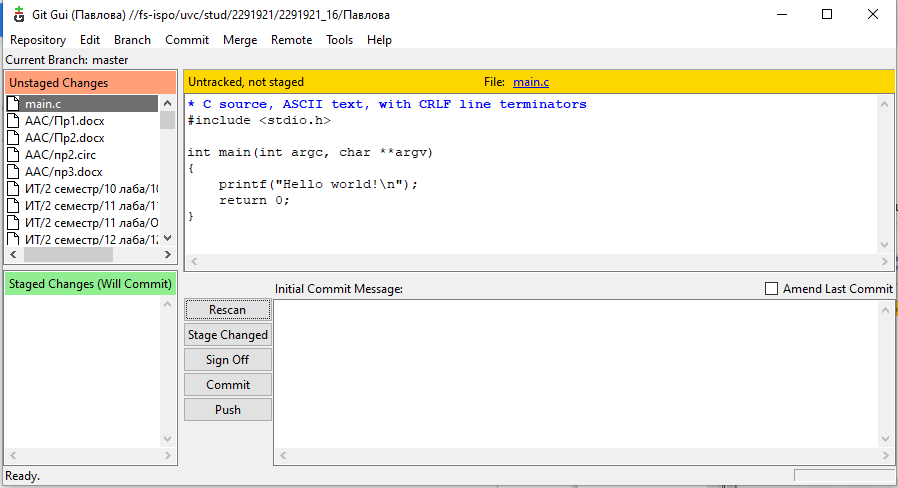


Рисунок 5-файл main.c

Чтобы добавить в этот файл фиксацию, щелкаю на иконку слева от имени файла. Файл переместился с Unstaged Changes (Изменено) панели на Staged Changes (Подготовлено) панель. Далее добавляю сообщение фиксации (commit message) и фиксирую изменения Commit (Сохранить) кнопкой (Рисунок 6).

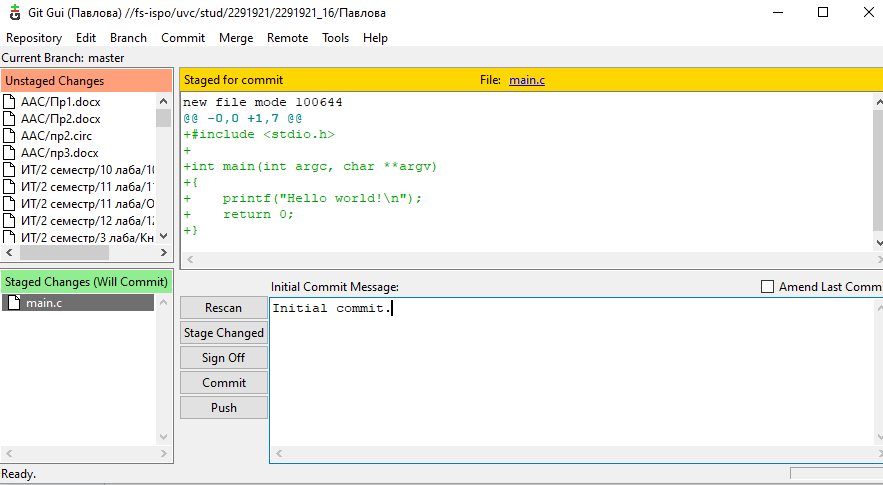


Рисунок 6-Фиксация изменений

Изменяю код в блокноте (Рисунок 7).

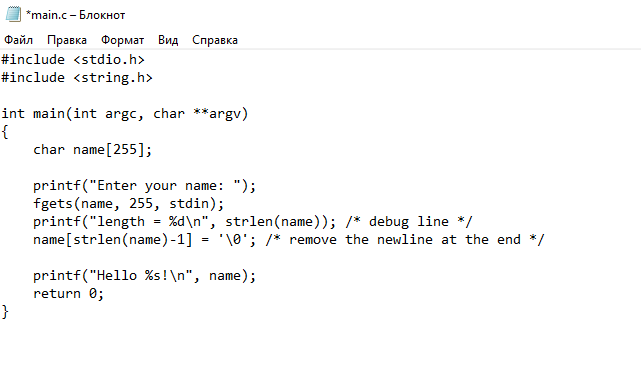


Рисунок 7-Изменённый код

Щелкаю Rescan для поиска изменений, изменения выделены красным (удаленные линии) или зеленым (добавленные линии). Щелкаю на иконку слева от файла, чтобы подготовить изменения к фиксации, файл будет перенесен в нижнее окно (Рисунок 8).

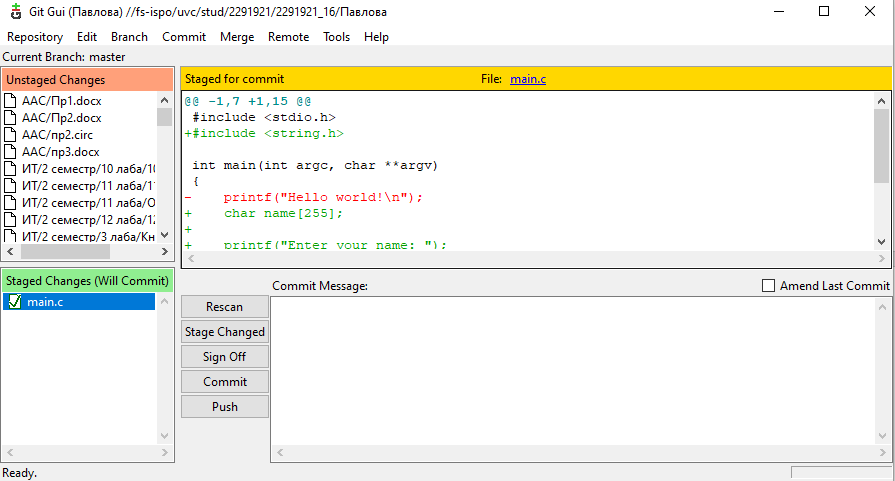


Рисунок 8-После изменения

Щелкаю правой кнопкой мыши на отладочной линии и выбираю Unstage Line From Commit (Рисунок 9).

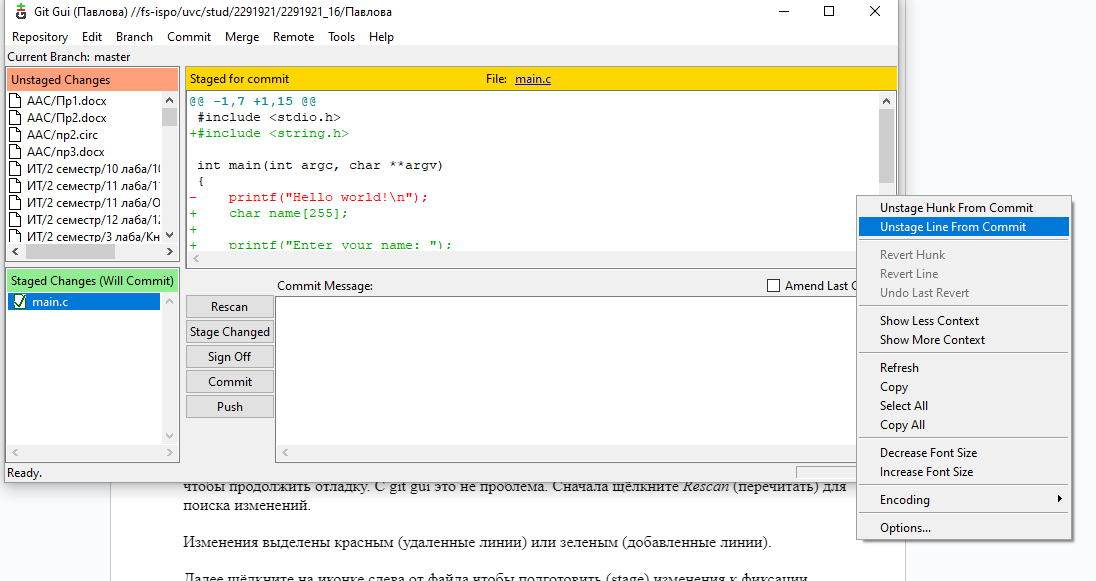


Рисунок 9 -Убрать строку из подготовленного

Теперь отладочная линия не была подготовлена к фиксации, в то время как остальные изменения были. Заполняю сообщения фиксации и фиксирую изменения щелкнув по Commit (Рисунок 10).

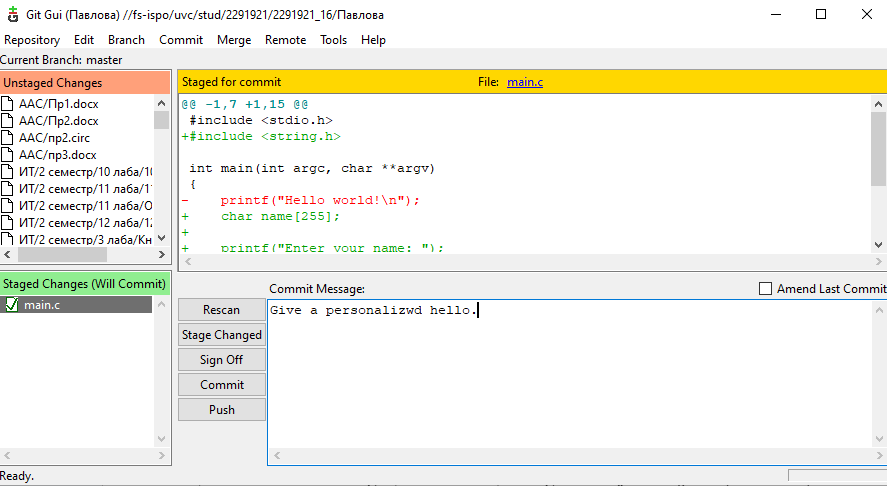


Рисунок 10-Фиксация изменений

Создаю ветку (branch). Выбираю Branch -> Create (Рисунок 11).

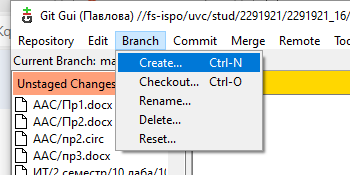


Рисунок 11-Создание ветки

Задаю имя ветки и нажимаю Create (Рисунок 12).

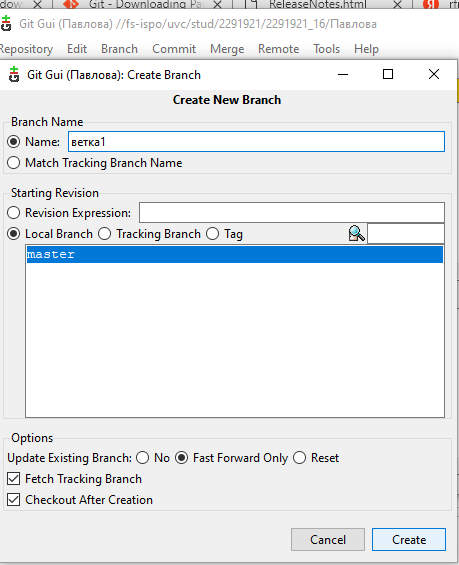


Рисунок 12-Название ветки

Делаю новые модификации в старом файле и создаю новый файл.с (Рисунок 13-14).

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 13-Модификация файла main.c

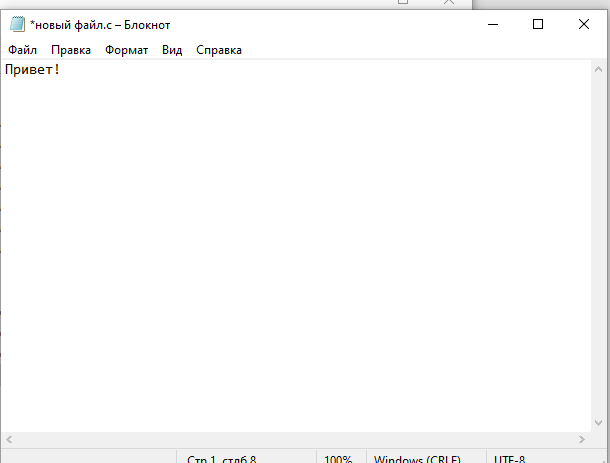


Рисунок 14-Новый файл

Фиксирую изменения (Рисунок 15).

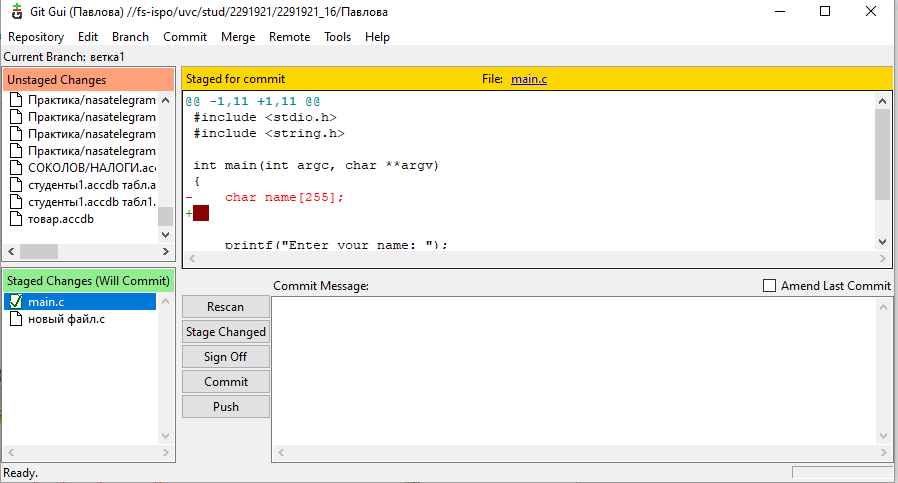


Рисунок 15-Фиксация изменений

Переключаюсь на ветку мастера Branch -> Checkout (Рисунок 16).

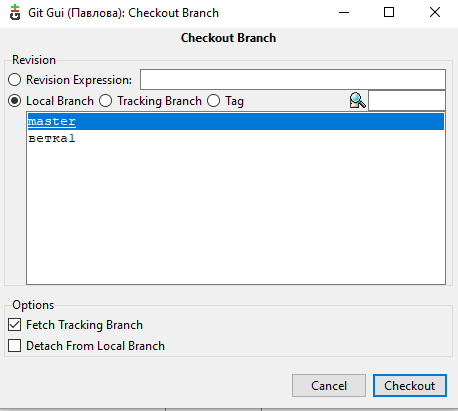


Рисунок 16-Переключение на ветrку master

Добавляю изменения в существующие файлы и создаю новый файл (Рисунок 17 - 19).

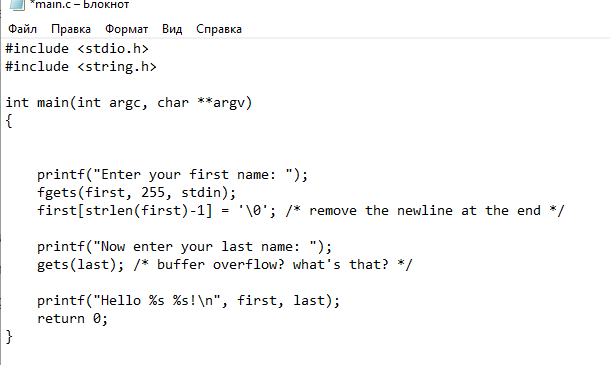


Рисунок 17- Файл main.c

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 18-новый файл.с

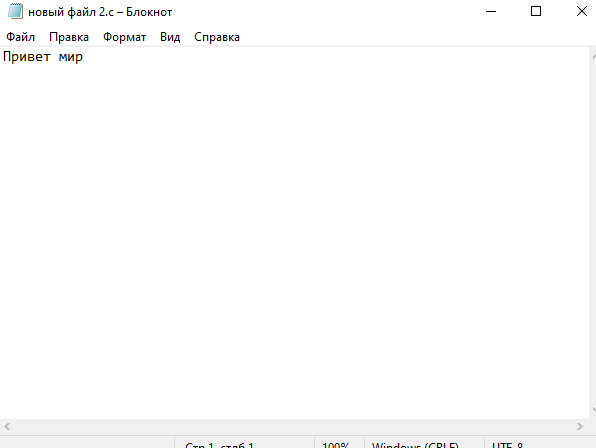


Рисунок 19-новый файл2.с

Фиксирую изменения (Рисунок 20).

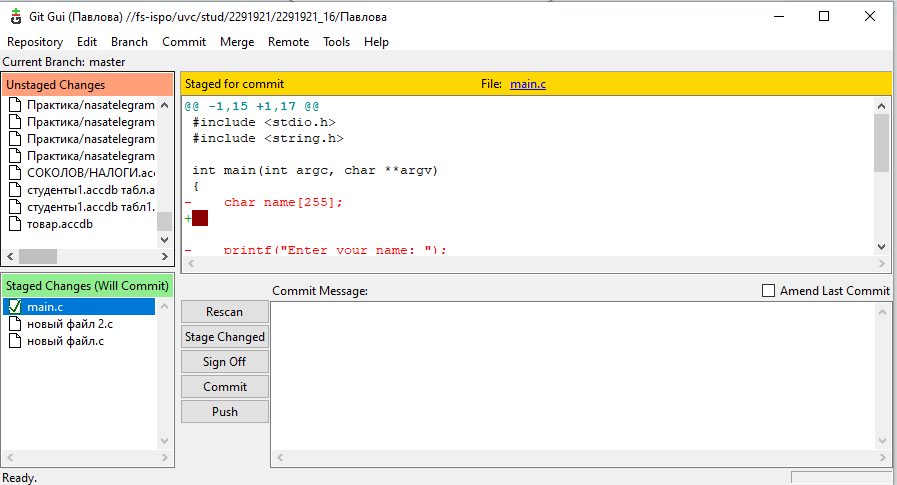


Рисунок 20-Фиксация изменений

Чтобы выполнить слияние, использую Merge -> Local Merge (Рисунок 21).

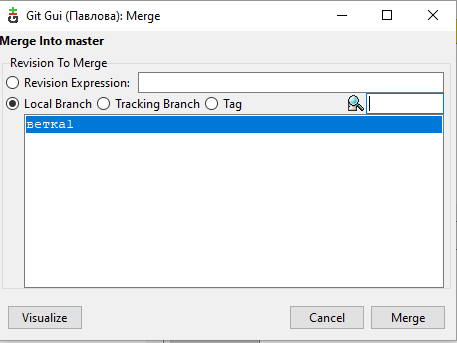


Рисунок 21-Merge into master

Так как две разных фиксации делали два разных изменения происходит конфликт (Рисунок 22).

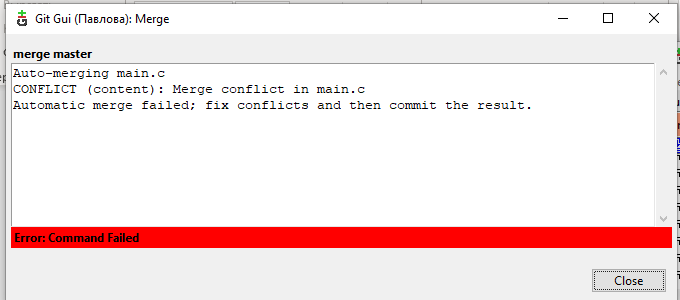


Рисунок 22-Конфликт

Разрешаю конфликт, используя блокнот (оставляю в тексте только нужный вариант) (Рисунок 23).

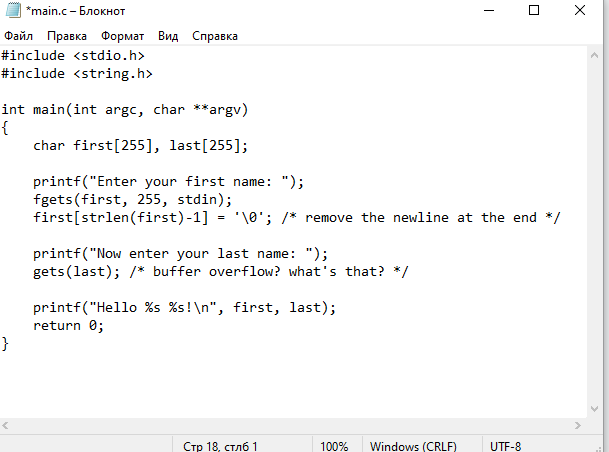


Рисунок 23-Правильный вариант

После разрешения конфликта подготавливаю изменения, щелкнув на иконку файла и фиксирую изменения (Рисунок 24).

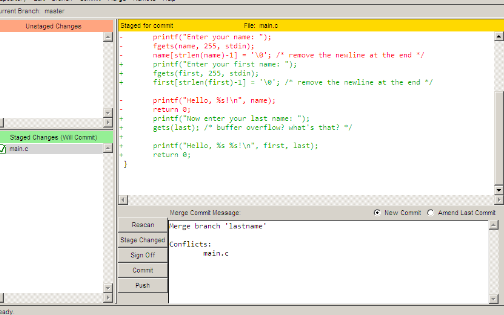


Рисунок 24-Фиксация слияния

Смотрю историю хранилища выбрав Repository -> Visualize All Branch History (Рисунок 25).

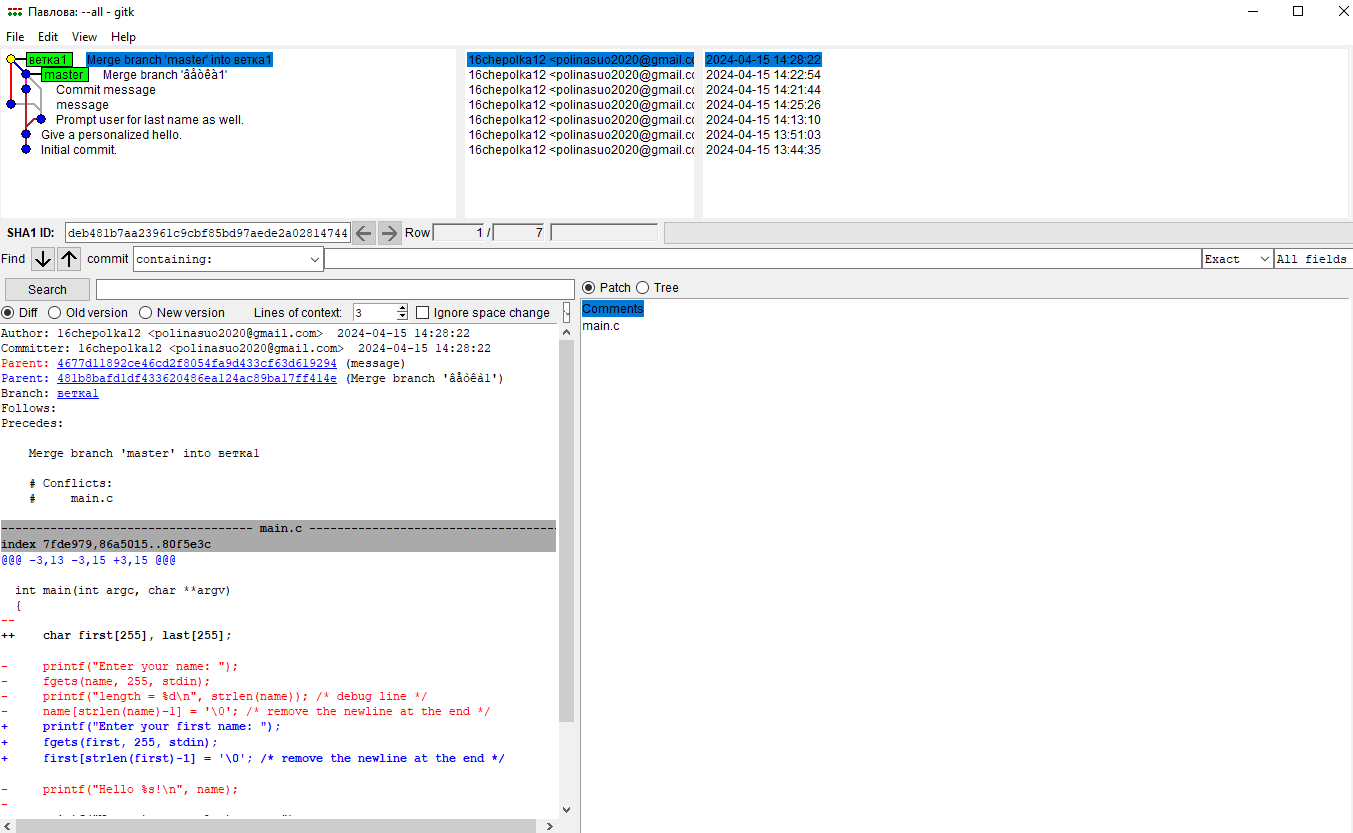


Рисунок 25-История хранилища

Выбираю Repository -> Browse master`s Files, из списка выбираю интересующий файл (Рисунок 26-27).

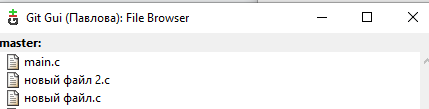


Рисунок 26-Список файлов master

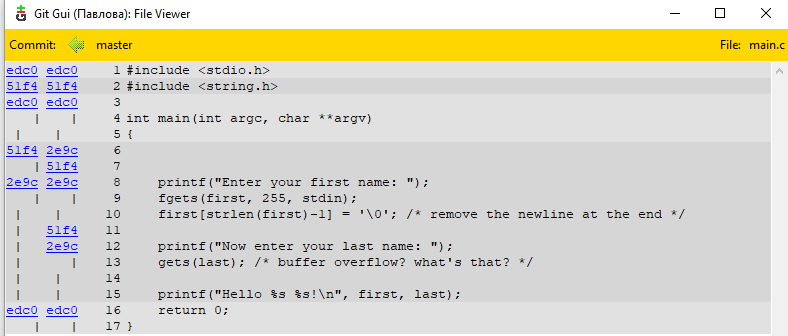


Рисунок 27-File Viewer

Откатываю к конкретной фиксации. Щелкаю правой кнопкой мыши и выбираю Reset master branch to here (Рисунок 28 - 30).

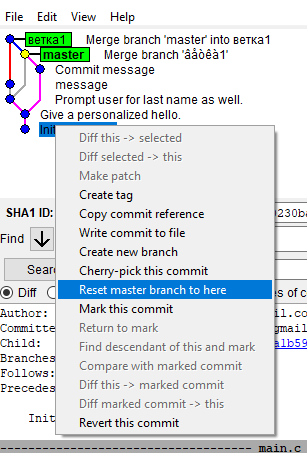


Рисунок 28-Reset master branch to here

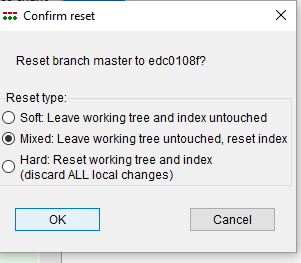


Рисунок 29-Confirm reset

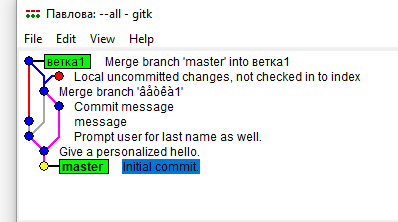


Рисунок 30-Результат

Создаю новый репозиторий на github (Рисунок 31).

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 31-Создание нового репозитория

Делаю репозиторий публичным и называю его (Рисунок 32).

Изображение выглядит как текст, снимок экрана, программное обеспечение, Операционная система

Автоматически созданное описание

Рисунок 32-Название репозитория

Далее выбираю в меню Внешние репозитории -> Отправить, указываю путь к репозиторию, нажимаю Push (Рисунок 33).

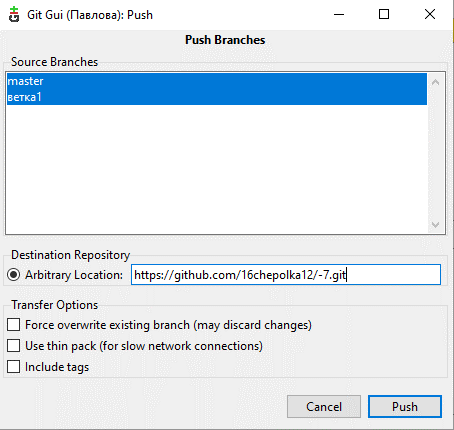


Рисунок 33-Push Branches

Захожу в свой github (Рисунок 34).

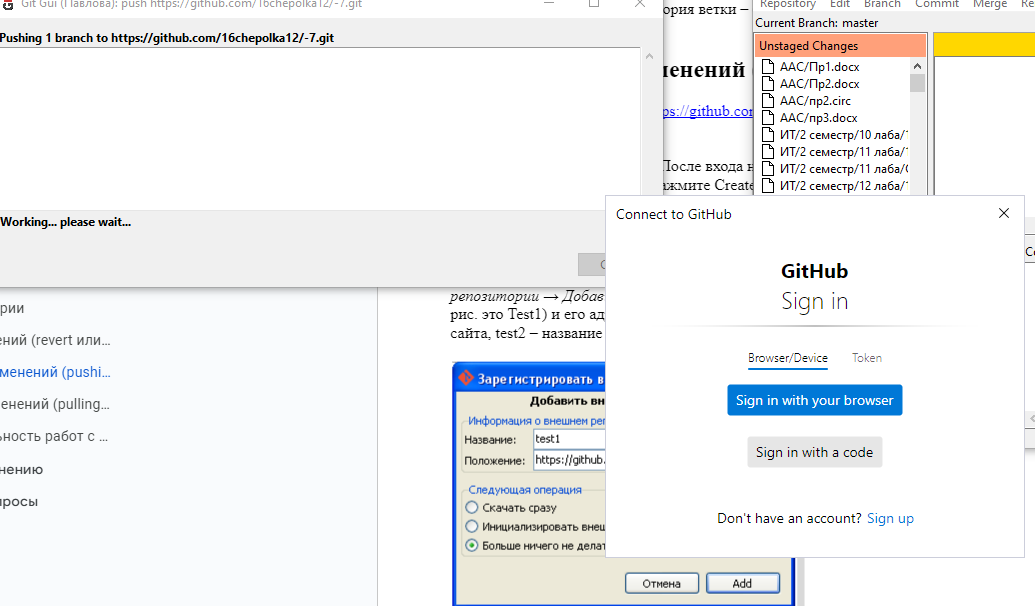


Рисунок 34-Connect to GitHub

Публикация изменений на удаленном сервере (Рисунок 35).

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок 35-Публикация изменений на удаленном сервере

Добавляю участника проекта к удаленному репозиторию (Рисунок 36).

Изображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

Автоматически созданное описание

Рисунок 36-Добавление участника проекта

Далее участник проекта получать данные из этого удаленного репозитория, изменяет их, фиксирует и отправляет их на удаленный репозиторий (Рисунок 37).

Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Автоматически созданное описание

Рисунок 37-Изменение в репозитории участником проекта

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описаниеПолучаю данные из репозитория и смотрю историю изменений (Рисунок 38).

Рисунок 38-История изменений