

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Донецкий национальный технический университет»

Факультет ИСП  
Кафедра ПИ им Л.П.Фельдмана

Лабораторная работа №2  
по курсу: «Профессиональная практика программной инженерии»  
по теме: «Работа с ветками»

Выполнил:  
ст. гр. ПИ-20  
Белоконь Л.О.

Проверили:  
Филипишин Д.А.

ДОНЕЦК – 2024

**Цель работы:** познакомиться с основами использования веток в системе Git.

## Ход работы

Сперва из ветки main были созданы 6 веток: code\_editor, program\_setting, search\_module, semantics\_analyzer, terminal, theme\_management (см. рис. 1). Затем, последовательно был осуществлён переход на каждую ветку с помощью команды «checkout <имя ветки>». В каждой ветке было совершено по 5 фиксаций изменений. Просмотреть историю вносимых изменений в ветку можно с помощью команды «log --pretty=format:"(%h)%ad|%s%d[%an]" --graph -date=short» (см. рис. 2-7).

```
MINGW64 /c/GitRepos/color-prog-pad (main)
$ git branch
  code_editor
* main
  program_setting
  search_module
  semantics_analyzer
  terminal
  theme_management
```

Рисунок 1 — Вывод списка веток

```
PS C:\GitRepos\color-prog-pad> git log --pretty=format:"(%h)%ad|%s%d[%an]" --graph --date=short
* \76f812b)2024-03-10|Добавлен обработчик нажатия клавиши Tab (HEAD -> code_editor)[Leva Belokon]\
* \f0a83cc)2024-03-10|Внесены изменения в свойства и методы класса CodeEditor[Leva Belokon]\
* \8d52bf2)2024-03-09|Реализованы основные методы класса CodeEditor[Leva Belokon]\
* \08e8022)2024-03-09|Объявлены основные методы класса CodeEditor[Leva Belokon]\
* \4e7412b)2024-03-09|Создано объявление класса CodeEditor в файле CodeEditor.cs[Leva Belokon]\
* \1c15d9a)2024-03-05|Исправлен отчёт к 1-й лр. (origin/main, origin/HEAD, main)[Leva Belokon]\
* \8271043)2024-03-05|Загружен отчёт 1-й лр.[Leva Belokon]\
* \eb53bb3)2024-03-05|Создана структура репозитория. Добавлено описание к разрабатываемой программе.[Leva Belokon]\
* \585ad09)2024-03-05|Initial commit[leva-prava]\
```

Рисунок 2 — Вывод лога для ветки code\_editor

```

PS C:\GitRepos\color-prog-pad> git checkout program_setting
Switched to branch 'program_setting'
PS C:\GitRepos\color-prog-pad> git log --pretty=format:"(%h)%ad|%s%d[%an]" --graph --date=short
* \(\cb90a6d)2024-03-10|В класс SettingM добавлена реализация метода GetSetting (HEAD -> program_setting)[Leva Belokon]\
* \(\0242911)2024-03-10|В лкасс SettingM добавлена реализация метода SetSetting[Leva Belokon]\
* \(\16bf565)2024-03-10|В класс SettingM добавлено объявление основных методов[Leva Belokon]\
* \(\eca2e5b)2024-03-10|В класс SettingM добавлено свойство settings[Leva Belokon]\
* \(\d85362c)2024-03-10|Создан класс SettingM[Leva Belokon]\
* \(\1c15d9a)2024-03-05|Исправлен отчёт к 1-й лр. (origin/main, origin/HEAD, main)[Leva Belokon]\
* \(\8271043)2024-03-05|Загружен отчёт 1-й лр.[Leva Belokon]\
* \(\eb53bb3)2024-03-05|Создана структура репозитория. Добавлено описание к разрабатываемой программе.[Leva Belokon]\
* \(\585ad09)2024-03-05|Initial commit[leva-prava]\

```

Рисунок 3 — Вывод лога для ветки program\_setting

```

PS C:\GitRepos\color-prog-pad> git checkout search_module
Switched to branch 'search_module'
PS C:\GitRepos\color-prog-pad> git log --pretty=format:"(%h)%ad|%s%d[%an]" --graph --date=short
* \(\a021842)2024-03-10|Внесены изменения в класс SearchM. Добавлены методы DisplayContext и CountOccurrences (HEAD -> search_module)[Leva Belokon]\
* \(\8ea88d0)2024-03-10|Внесены изменения в класс SearchM. Добавлен метод ReplaceUsingRegex[Leva Belokon]\
* \(\009d508)2024-03-10|Реализованы основные методы класса SearchM[Leva Belokon]\
* \(\62d9a4a)2024-03-10|Объявлены основные методы класса SearchM[Leva Belokon]\
* \(\9fb9072)2024-03-10|Создан класс SearchM[Leva Belokon]\
* \(\1c15d9a)2024-03-05|Исправлен отчёт к 1-й лр. (origin/main, origin/HEAD, main)[Leva Belokon]\
* \(\8271043)2024-03-05|Загружен отчёт 1-й лр.[Leva Belokon]\
* \(\eb53bb3)2024-03-05|Создана структура репозитория. Добавлено описание к разрабатываемой программе.[Leva Belokon]\
* \(\585ad09)2024-03-05|Initial commit[leva-prava]\

```

Рисунок 4 — Вывод лога для ветки search\_module

```

PS C:\GitRepos\color-prog-pad> git checkout semantics_analyzer
Switched to branch 'semantics_analyzer'
PS C:\GitRepos\color-prog-pad> git log --pretty=format:"(%h)%ad|%s%d[%an]" --graph --date=short
* \(\8705f96)2024-03-10|Внесены изменения в структуру класса SemanticsAnalyzer (HEAD -> semantics_analyzer)[Leva Belokon]\
* \(\aca70b1)2024-03-10|Изменена реализация метода AnalyzeCode класса SemanticsAnalyzer[Leva Belokon]\
* \(\d36f833)2024-03-10|Выполнена реализация основных методов класса SemanticsAnalyzer[Leva Belokon]\
* \(\7cddf6d)2024-03-10|Объявлены основные свойства и методы класса SemanticsAnalyzer[Leva Belokon]\
* \(\f4c6ac2)2024-03-10|Создан класс SemanticsAnalyzer[Leva Belokon]\
* \(\1c15d9a)2024-03-05|Исправлен отчёт к 1-й лр. (origin/main, origin/HEAD, main)[Leva Belokon]\
* \(\8271043)2024-03-05|Загружен отчёт 1-й лр.[Leva Belokon]\
* \(\eb53bb3)2024-03-05|Создана структура репозитория. Добавлено описание к разрабатываемой программе.[Leva Belokon]\
* \(\585ad09)2024-03-05|Initial commit[leva-prava]\

```

Рисунок 5 — Вывод лога для ветки semantics\_analyzer

```

PS C:\GitRepos\color-prog-pad> git checkout terminal
Switched to branch 'terminal'
PS C:\GitRepos\color-prog-pad> git log --pretty=format:"(%h)%ad|%s%d[%an]" --graph --date=short
* \9c74935)2024-03-10|Добавленна функция вывода истории команд для класса Terminal (HEAD -> terminal)[Leva Belokon]\
* \e4ec4ce)2024-03-10|Добавлен метод DisplayHelp в класс Terminal[Leva Belokon]\
* \e4a7159)2024-03-10|Внесены изменения в метод ProcessCommand[Leva Belokon]\
* \1b767fe)2024-03-10|Выполнена реализация основных методов класса Terminal[Leva Belokon]\
* \1fa4f4d)2024-03-10|Создан класс Terminal[Leva Belokon]\
* \1c15d9a)2024-03-05|Исправлен отчёт к 1-й лр. (origin/main, origin/HEAD, main)[Leva Belokon]\
* \8271043)2024-03-05|Загружен отчёт 1-й лр.[Leva Belokon]\
* \eb53bb3)2024-03-05|Создана структура репозитория. Добавлено описание к разрабатываемой программе.[Leva Belokon]\
* \585ad09)2024-03-05|Initial commit[leva-prava]\

```

Рисунок 6 — Вывод лога для ветки terminal

```

PS C:\GitRepos\color-prog-pad> git checkout theme_management
Switched to branch 'theme_management'
PS C:\GitRepos\color-prog-pad> git log --pretty=format:"(%h)%ad|%s%d[%an]" --graph --date=short
* \c8ac027)2024-03-10|Изменена структура клсса ThemeManagementM (HEAD -> theme_management)[Leva Belokon]\
* \fc06d35)2024-03-10|Выполнена реализация основных методов класса ThemeManagementM[Leva Belokon]\
* \ccc8408)2024-03-10|Добавлено объявление основных свойств и методов класса ThemeManagementM[Leva Belokon]\
* \1bd0b30)2024-03-10|В класс ThemeManagement добавлено перечисление Theme[Leva Belokon]\
* \6f9d8c3)2024-03-10|Создан класс ThemeManagement[Leva Belokon]\
* \1c15d9a)2024-03-05|Исправлен отчёт к 1-й лр. (origin/main, origin/HEAD, main)[Leva Belokon]\
* \8271043)2024-03-05|Загружен отчёт 1-й лр.[Leva Belokon]\
* \eb53bb3)2024-03-05|Создана структура репозитория. Добавлено описание к разрабатываемой программе.[Leva Belokon]\
* \585ad09)2024-03-05|Initial commit[leva-prava]\

```

Рисунок 7 — Вывод лога для ветки theme\_management

Затем после того, как была завершена работа во всех ветках, зафиксированные изменения были отправлены на удалённый репозиторий с помощью команды «push origin --all» (см. рис. 8-9).

```

MINGW64 /c/GitRepos/color-prog-pad (main)
$ git push origin --all
info: please complete authentication in your browser...
Enumerating objects: 122, done.
Counting objects: 100% (122/122), done.
Delta compression using up to 4 threads
Compressing objects: 100% (120/120), done.
Writing objects: 100% (120/120), 16.73 KiB | 535.00 KiB/s, done.
Total 120 (delta 33), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (33/33), done.
To https://github.com/leva-prava/color-prog-pad
* [new branch]      code_editor -> code_editor
* [new branch]      program_setting -> program_setting
* [new branch]      search_module -> search_module
* [new branch]      semantics_analyzer -> semantics_analyzer
* [new branch]      terminal -> terminal
* [new branch]      theme_management -> theme_management

```

Рисунок 8 — Отправка изменений на удалённый репозиторий

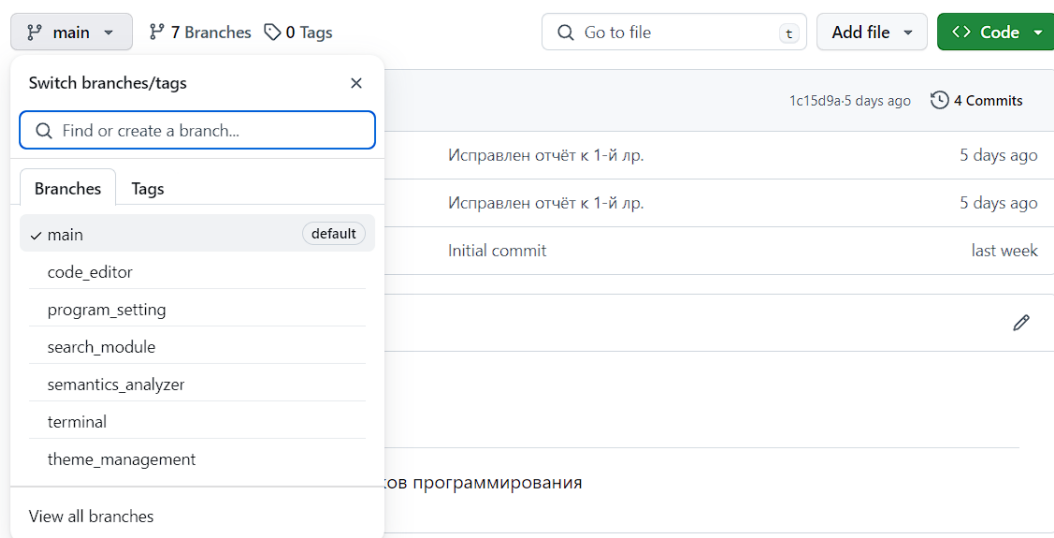


Рисунок 9 — Просмотр списка веток через сервис GitHub

Для того, чтобы продемонстрировать разрешение конфликта слияния веток, создадим конфликтную ситуацию: в ветке `main` создадим копию файла `Terminal.cs` из ветки `Terminal`, изменив некоторые строки. После выполнения команды «`git merge terminal`» на консоли будет выведено сообщение о том, что

git не смог решить конфликт слияния автоматически (см. рис. 10), после чего редактор VS Code предложит в удобном формате выбрать для каждой конфликтной ситуации нужный вариант строк (см. рис. 10-13).

```
MINGW64 /c/GitRepos/color-prog-pad (main)
$ git merge terminal
Auto-merging Program/Terminal.cs
CONFLICT (add/add): Merge conflict in Program/Terminal.cs
Automatic merge failed; fix conflicts and then commit the result.
```

Рисунок 10 — Сообщение о конфликте слияния

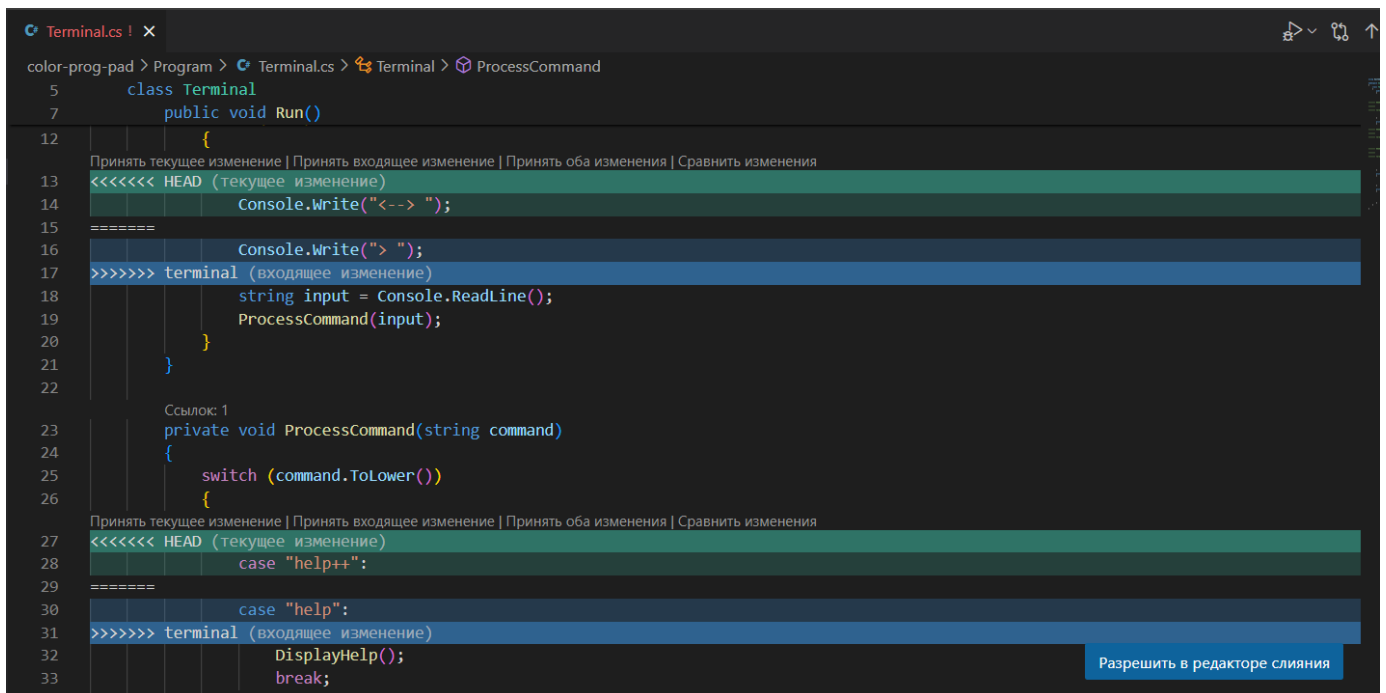


Рисунок 11 — Окно устранения конфликтов слияния редактора VS Code

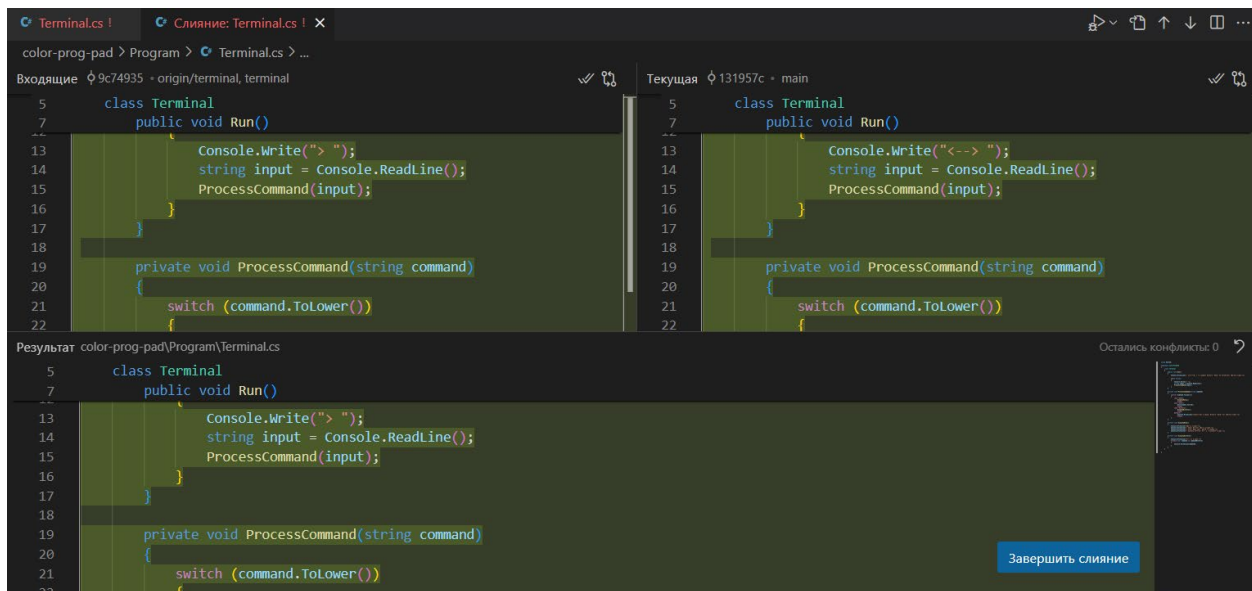


Рисунок 12 — Окно завершения слияния редактора VS Code (1-я конфликтная ситуация)

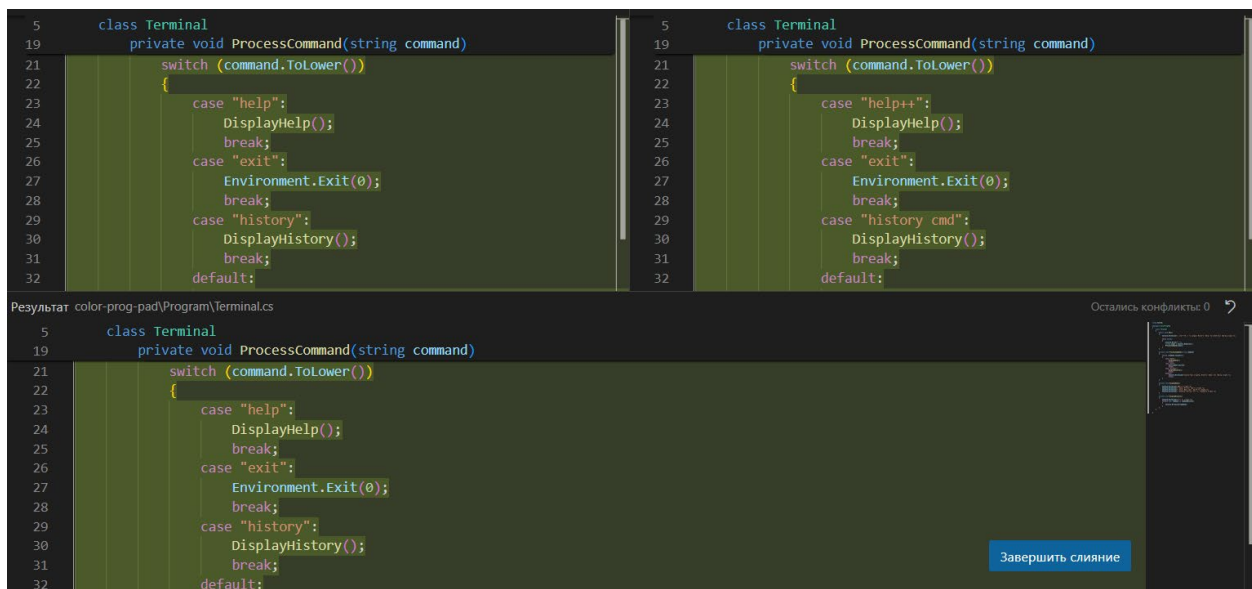


Рисунок 13 — Окно завершения слияния редактора VS Code (2-я конфликтная ситуация)

После того, как будут выбраны нужные варианты конфликтующих строк, необходимо зафиксировать изменения. В редакторе VS Code это предлагается сделать с помощью нажатия кнопки (см. рис. 14-15).

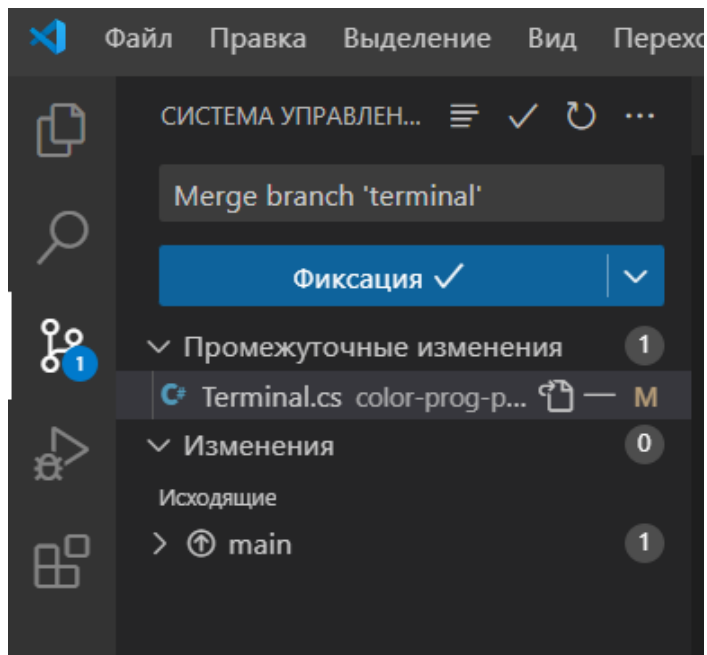


Рисунок 14 — Фиксация разрешенного конфликта слияния

```
PS C:\GitRepos\color-prog-pad> git log --pretty=format:"(%h)%ad|%s%d[%an]" --graph --date=short
* \(\18105d7)2024-03-10|Merge branch 'terminal' (HEAD -> main)[Leva Belokon]\
|
| * \(\9c74935)2024-03-10|Добавленна функция вывода истории команд для класса Terminal (origin/terminal, terminal)[Leva Belokon]\
| * \(\e4ec4ce)2024-03-10|Добавлен метод DisplayHelp в класс Terminal[Leva Belokon]\
| * \(\e4a7159)2024-03-10|Внесены изменения в метод ProcessCommand[Leva Belokon]\
| * \(\1b767fe)2024-03-10|Выполнена реализация основных методов класса Terminal[Leva Belokon]\
| * \(\1fa4f4d)2024-03-10|Создан класс Terminal[Leva Belokon]\
| * \(\131957c)2024-03-10|В ветке main создан класс Terminal[Leva Belokon]\
|/
* \(\1c15d9a)2024-03-05|Исправлен отчёт к 1-й лр. (origin/main, origin/HEAD)[Leva Belokon]\
* \(\8271043)2024-03-05|Загружен отчёт 1-й лр.[Leva Belokon]\
* \(\eb53bb3)2024-03-05|Создана структура репозитория. Добавлено описание к разрабатываемой программе.[Leva Belokon]\
* \(\585ad09)2024-03-05|Initial commit[leva-prava]\
PS C:\GitRepos\color-prog-pad>
```

Рисунок 15 — Лог ветки main после слияния с веткой terminal

Затем с помощью команды «git merge -m “сообщение” <имя ветки> » было выполнено последовательное слияние остальных веток с веткой main. На рисунке 16 представлен лог ветки main после слияния со всеми второстепенными ветками.



```

PS C:\GitRepos\color-prog-pad> git log --pretty=format:"(%h)%ad%s%d[an]" --graph --date=short
*   \6c50ab7)2024-03-10|Слияние ветки [theme_management] с [main] (HEAD -> main)[Leva Belokon]\
|
| *   \c8ac027)2024-03-10|Изменена структура клсса ThemeManagementM (origin/theme_management, theme_management)[Leva Belokon]\
| *   \fc06d35)2024-03-10|Выполнена реализация основных методов класса ThemeManagementM[Leva Belokon]\
| *   \ccc8408)2024-03-10|Добавлено объявление основных свойств и методов класса ThemeManagementM[Leva Belokon]\
| *   \1bd0b30)2024-03-10|В класс ThemeManagement добавлено перечисление Theme[Leva Belokon]\
| *   \6f9d8c3)2024-03-10|Создан класс ThemeManagement[Leva Belokon]\
| *   \6e08c9c)2024-03-10|Слияние ветки [semantics_analyzer] с [main][Leva Belokon]\
|
| *   \8705f96)2024-03-10|Внесены изменения в структуру класса SemanticsAnalyzer (origin/semantics_analyzer, semantics_analyzer)[Leva Belokon]\
| *   \aca70b1)2024-03-10|Изменена реализация метода AnalyzeCode класса SemanticsAnalyzer[Leva Belokon]\
| *   \d36f833)2024-03-10|Выполнена реализация основных методов класса SemanticsAnalyzer[Leva Belokon]\
| *   \7cddf6d)2024-03-10|Объявлены основные свойства и методы класса SemanticsAnalyzer[Leva Belokon]\
| *   \f4c6ac2)2024-03-10|Создан класс SemanticsAnalyzer[Leva Belokon]\
|
| *   \5414597)2024-03-10|Слияние ветки [search_module] с [main][Leva Belokon]\
|
| *   \a021842)2024-03-10|Внесены изменения в класс SearchM. Добавлены методы DisplayContext и CountOccurrences (origin/search_module, search_modul
e)[Leva Belokon]\
| *   \8ea88d0)2024-03-10|Внесены изменения в класс SearchM. Добавлен метод ReplaceUsingRegex[Leva Belokon]\
| *   \009d508)2024-03-10|Реализованы основные методы класса SearchM[Leva Belokon]\
| *   \62d9a4a)2024-03-10|Объявлены основные методы класса SearchM[Leva Belokon]\
| *   \9fb9072)2024-03-10|Создан класс SearchM[Leva Belokon]\
|
| *   \eb9c28e)2024-03-10|Слияние ветки [program_setting] с [main][Leva Belokon]\
|
| *   \cb90a6d)2024-03-10|В класс SettingM добавлена реализация метода GetSetting (origin/program_setting, program_setting)[Leva Belokon]\
| *   \0242911)2024-03-10|В ласс SettingM добавлена реализация метода SetSetting[Leva Belokon]\
| *   \16bf565)2024-03-10|В класс SettingM добавлено объявление основных методов[Leva Belokon]\
| *   \eca2e5b)2024-03-10|В класс SettingM добавлено свойство settings[Leva Belokon]\
| *   \d85362c)2024-03-10|Создан класс SettingM[Leva Belokon]\
|
| *   \1f6006f)2024-03-10|Merge branch 'code_editor' Слияние ветки [code_editor] с веткой [main][Leva Belokon]\
|
| *   \76f812b)2024-03-10|Добавлен обработчик нажатия клавиши Tab (origin/code_editor, code_editor)[Leva Belokon]\
| *   \f0a83cc)2024-03-10|Внесены изменения в свойства и методы класса CodeEditor[Leva Belokon]\
| *   \8d52bf2)2024-03-09|Реализованы основные методы класса CodeEditor[Leva Belokon]\
| *   \08e8022)2024-03-09|Объявлены основные методы класса CodeEditor[Leva Belokon]\
| *   \4e7412b)2024-03-09|Создано объявление класса CodeEditor в файле CodeEditor.cs[Leva Belokon]\

```

Рисунок 16 — Лог ветки main после слияния с второстепенными ветками

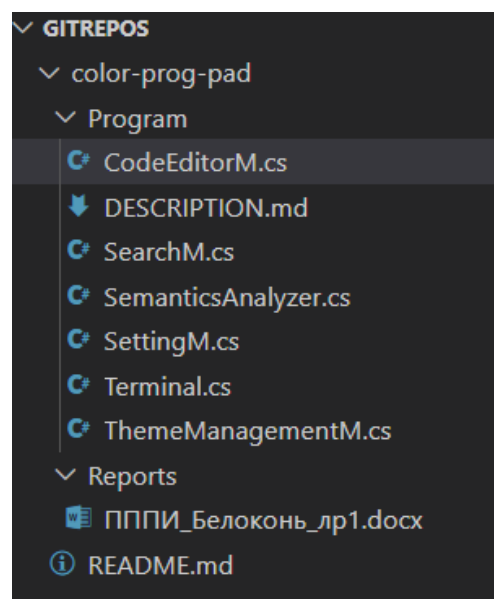


Рисунок 17 — Дерево файлов ветки main после слияния

После слияния, каждая из веток была удалена командой «branch -d» (см. рис. 18-19).

```
MINGW64 /c/GitRepos/color-prog-pad (main)
$ git branch -d code_editor
Deleted branch code_editor (was 76f812b).

MINGW64 /c/GitRepos/color-prog-pad (main)
$ git branch -d program_setting
Deleted branch program_setting (was cb90a6d).

MINGW64 /c/GitRepos/color-prog-pad (main)
$ git branch -d search_module
Deleted branch search_module (was a021842).

MINGW64 /c/GitRepos/color-prog-pad (main)
$ git branch -d semantics_analyzer
Deleted branch semantics_analyzer (was 8705f96).

MINGW64 /c/GitRepos/color-prog-pad (main)
$ git branch -d terminal
Deleted branch terminal (was 9c74935).

MINGW64 /c/GitRepos/color-prog-pad (main)
$ git branch -d theme_management
Deleted branch theme_management (was c8ac027).
```

Рисунок 18 — Процесс удаления второстепенных веток

```
MINGW64 /c/GitRepos/color-prog-pad (main)
$ git branch
* main
```

Рисунок 19 — Список веток

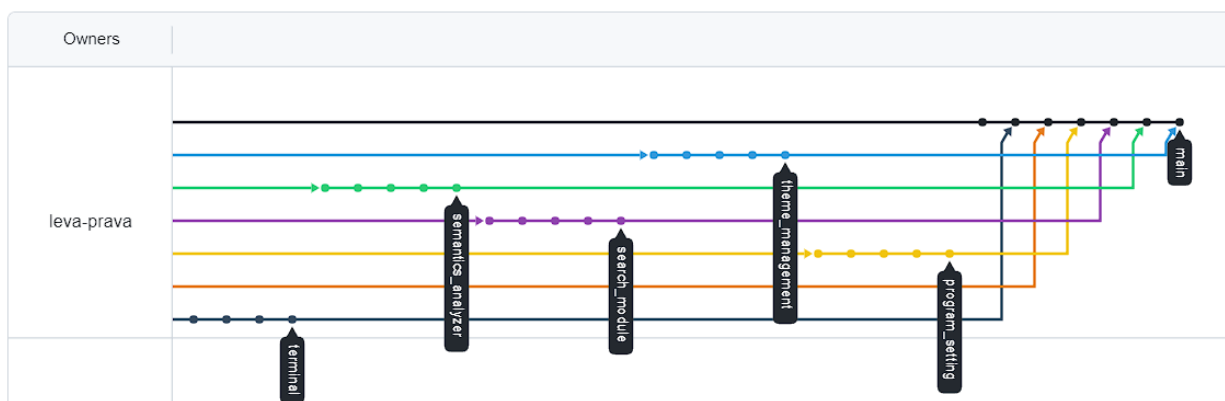


Рисунок 20 — График слияния веток