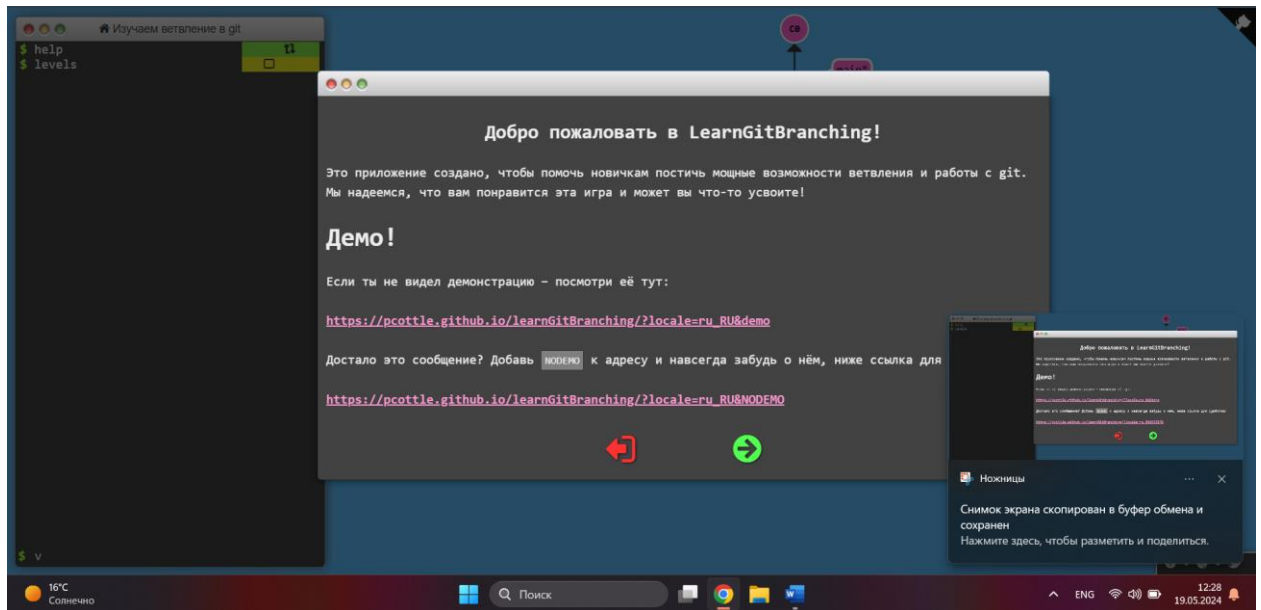
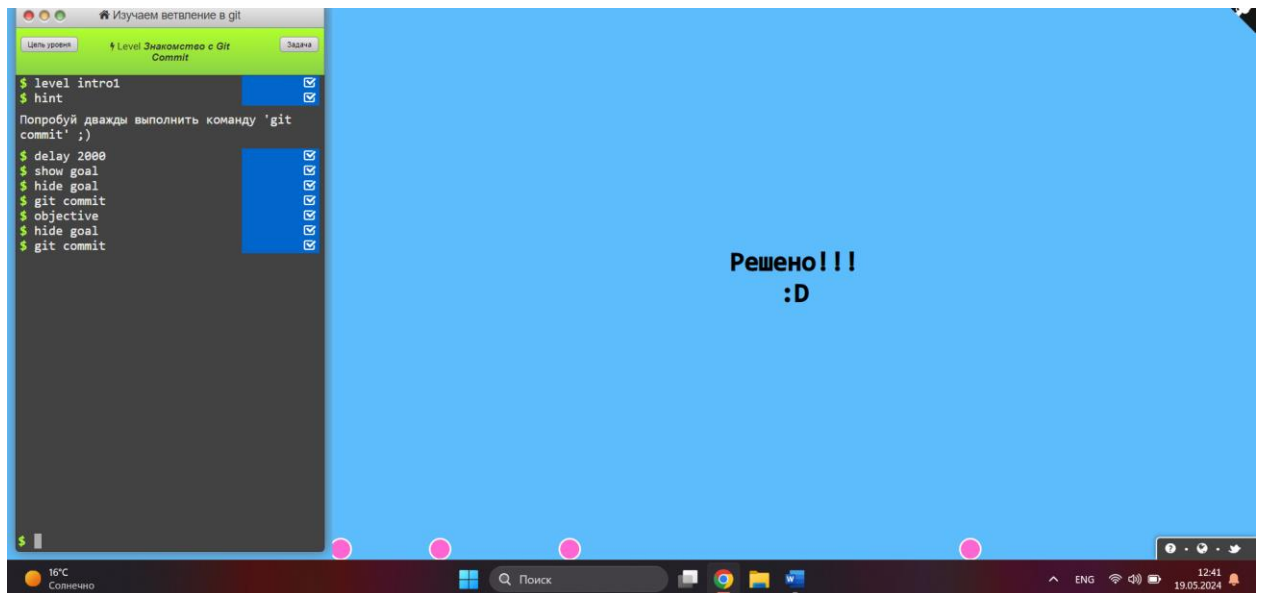


Маторкин Дмитрий Алексеевич 214В



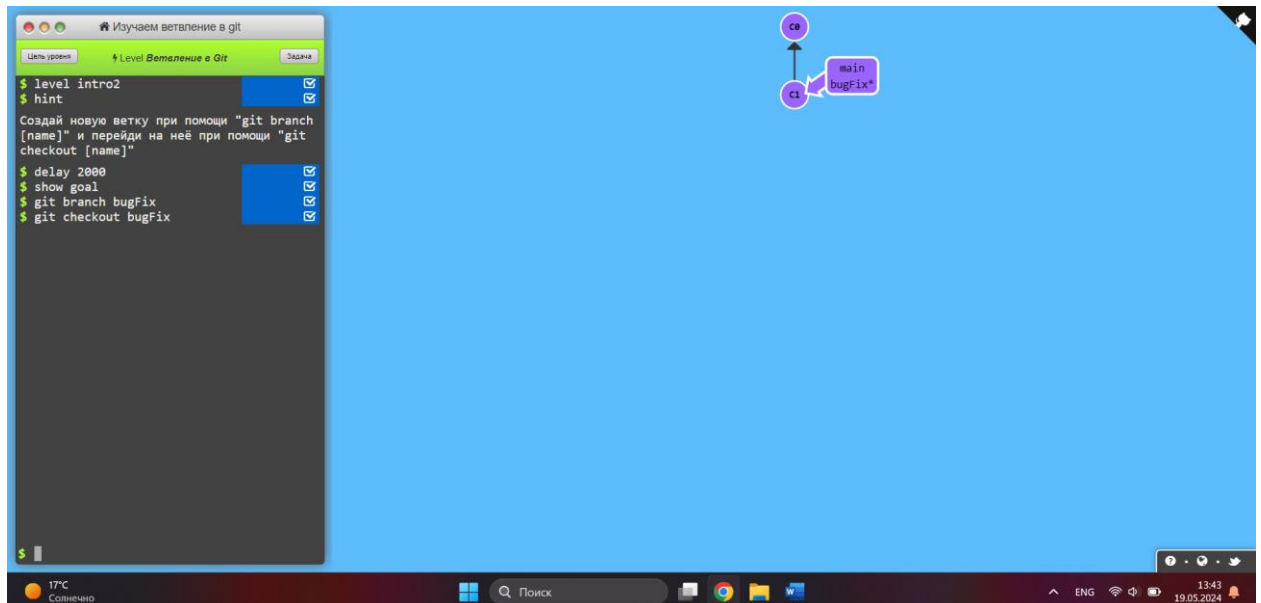
1.1

```
$ level intro1
$ hint
Попробуй дважды выполнить команду 'git commit' ;)
$ delay 2000
$ show goal
$ hide goal
$ git commit
$ objective
$ hide goal
$ git commit
```



1.2

```
$ level intro2
$ hint
Создай новую ветку при помощи "git branch [name]" и перейди на неё при помощи "git checkout [name]"
$ delay 2000
$ show goal
$ git branch bugFix
$ git checkout bugFix
```



1.3

```
$ level intro3
$ hint
```

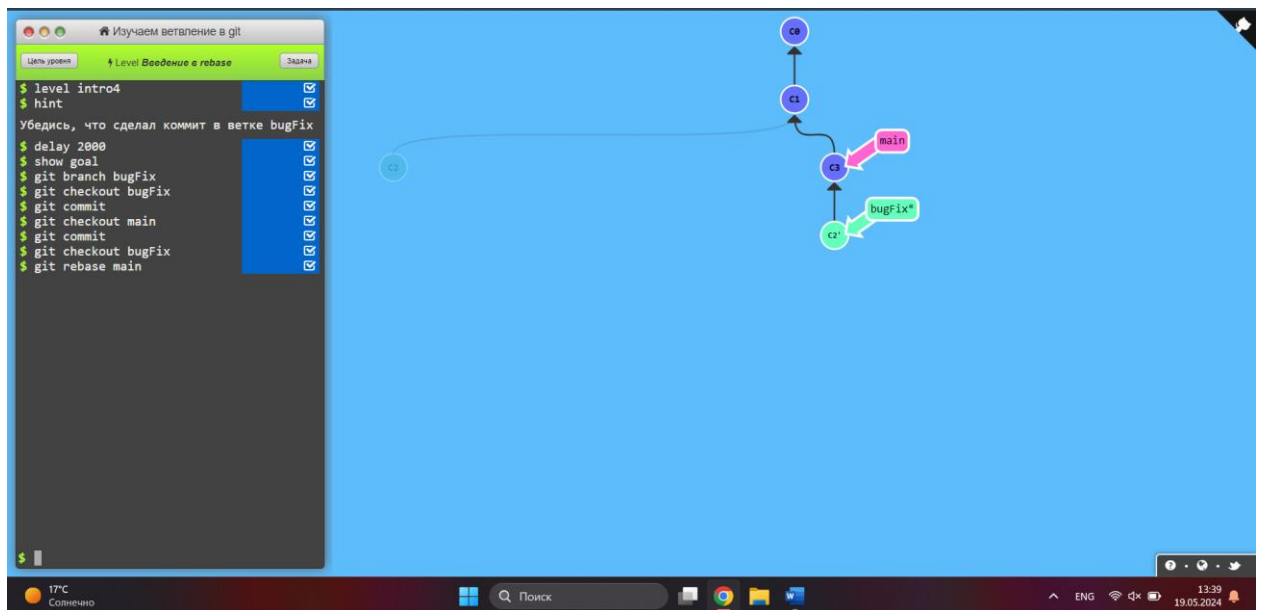
Не забудь делать коммиты в правильном порядке (сначала bugFix, потом main)

```
$ delay 2000
$ show goal
$ git branch bugFix
$ git checkout bugFix
$ git commit
$ git checkout main
$ git commit
$ git merge bugFix
```

The screenshot shows a Git tutorial interface. On the left is a terminal window titled "Изучаем ветвление в git" with a green header bar. It displays a list of tasks with checkboxes and a sequence of Git commands. The commands are: `$ level intro3`, `$ hint`, `$ delay 2000`, `$ show goal`, `$ git branch bugFix`, `$ git checkout bugFix`, `$ git commit`, `$ git checkout main`, `$ git commit`, and `$ git merge bugFix`. The first two tasks are checked, and the last two are not. To the right of the terminal is a commit graph on a light blue background. It shows a sequence of commits: `c0` (purple), `c1` (purple), `c2` (pink), and `c4` (pink). A green arrow labeled "bugFix" points from `c1` to `c2`. A pink arrow labeled "main" points from `c2` to `c4`. The graph illustrates the workflow of creating a branch, making changes, and merging them back to the main branch.

1.4

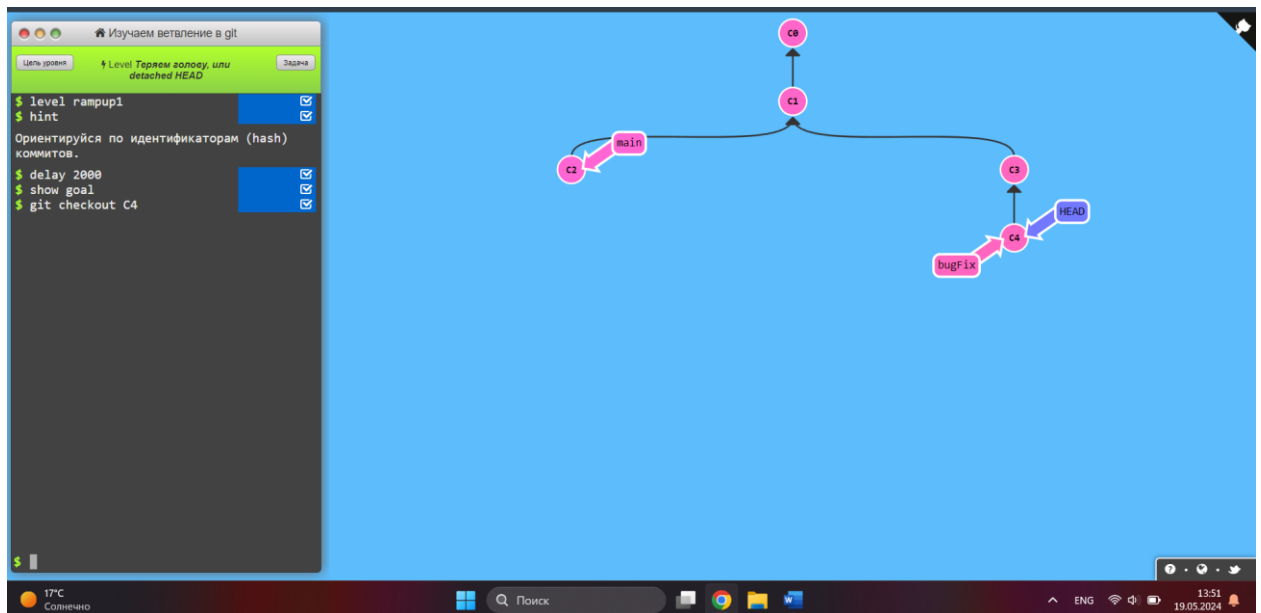
```
$ level intro4
$ hint
Убедись, что сделал коммит в ветке bugFix
$ delay 2000
$ show goal
$ git branch bugFix
$ git checkout bugFix
$ git commit
$ git checkout main
$ git commit
$ git checkout bugFix
$ git rebase main
```



2.1

```

$ level rampup1
$ hint
Ориентируйся по идентификаторам (hash) коммитов.
$ delay 2000
$ show goal
$ git checkout C4
  
```

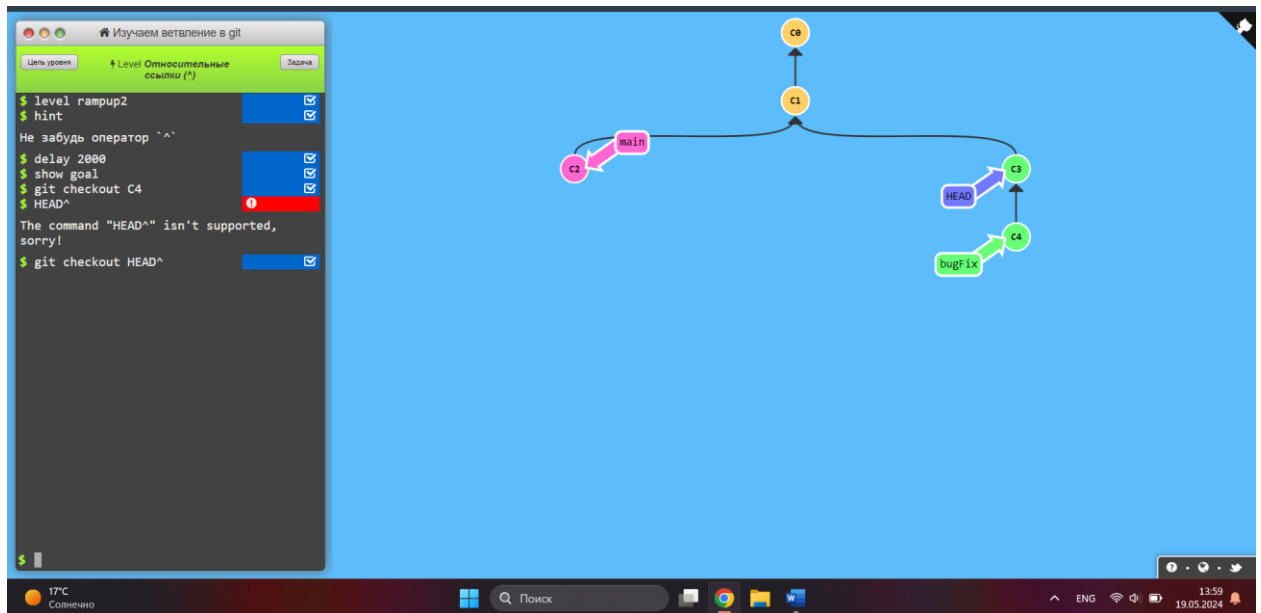


2.2

```

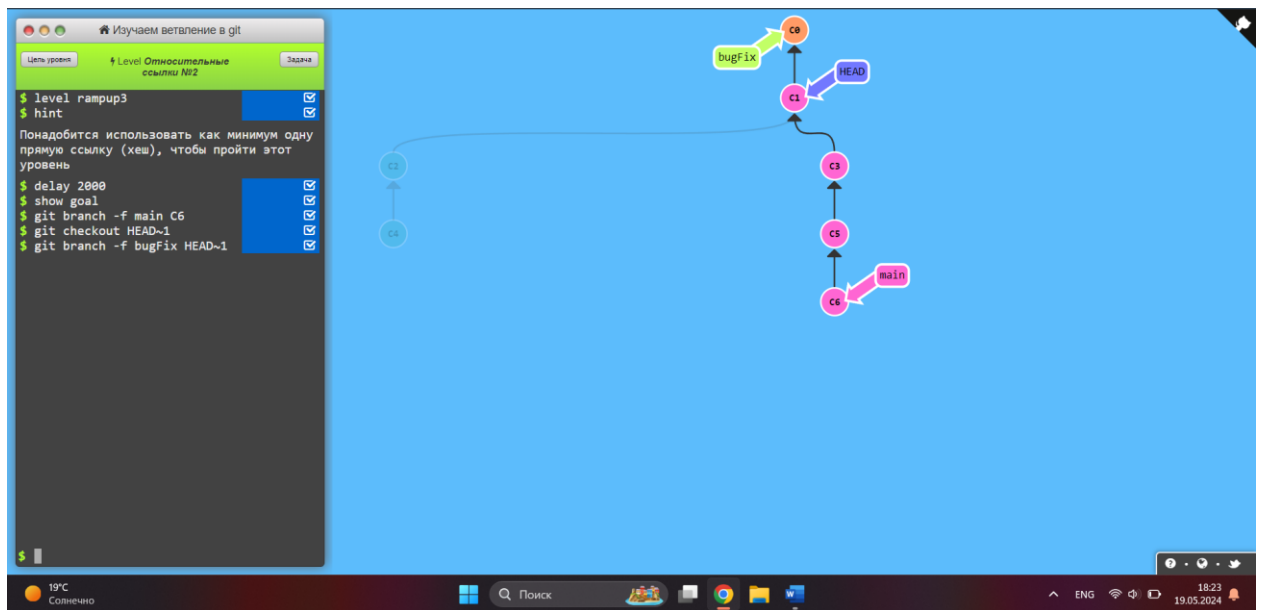
$ level rampup2
$ hint
Не забудь оператор `^^`
$ delay 2000
$ show goal
  
```

```
$ git checkout C4
$ git checkout HEAD^
```



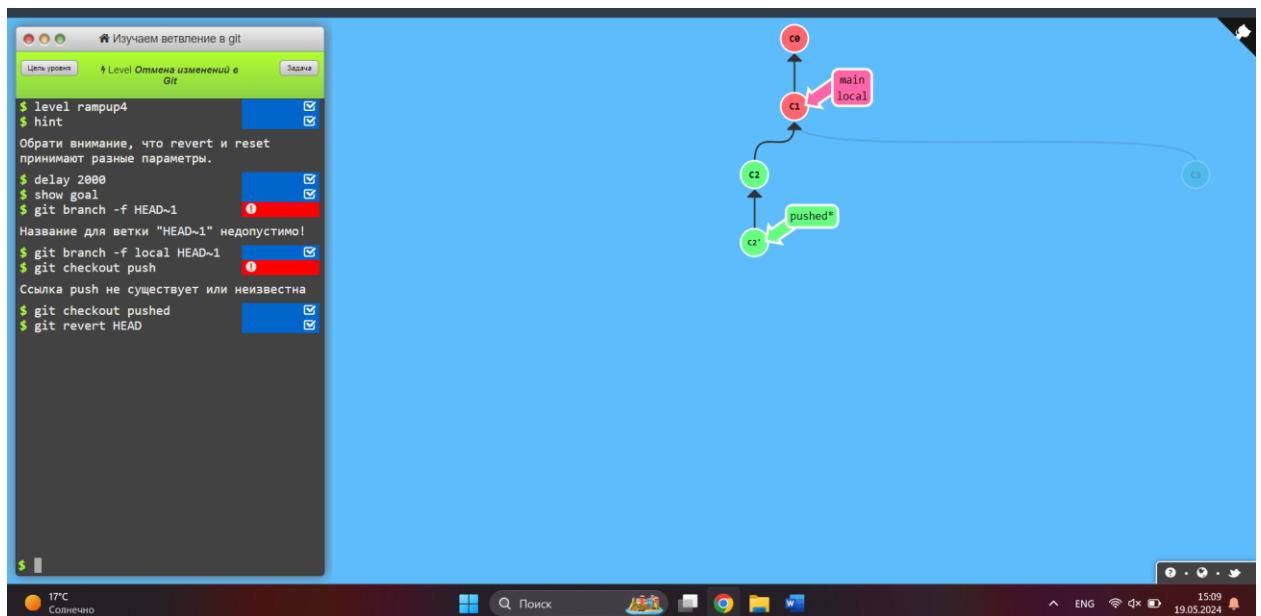
2.3

```
$ level rampup3
$ hint
Понадобится использовать как минимум одну прямую ссылку (хеш),
чтобы пройти этот уровень
$ delay 2000
$ show goal
$ git branch -f main C6
$ git checkout HEAD~1
$ git branch -f bugFix HEAD~1
```



2.4

```
$ git branch -f local HEAD~1
$ git checkout pushed
$ git revert HEAD
```



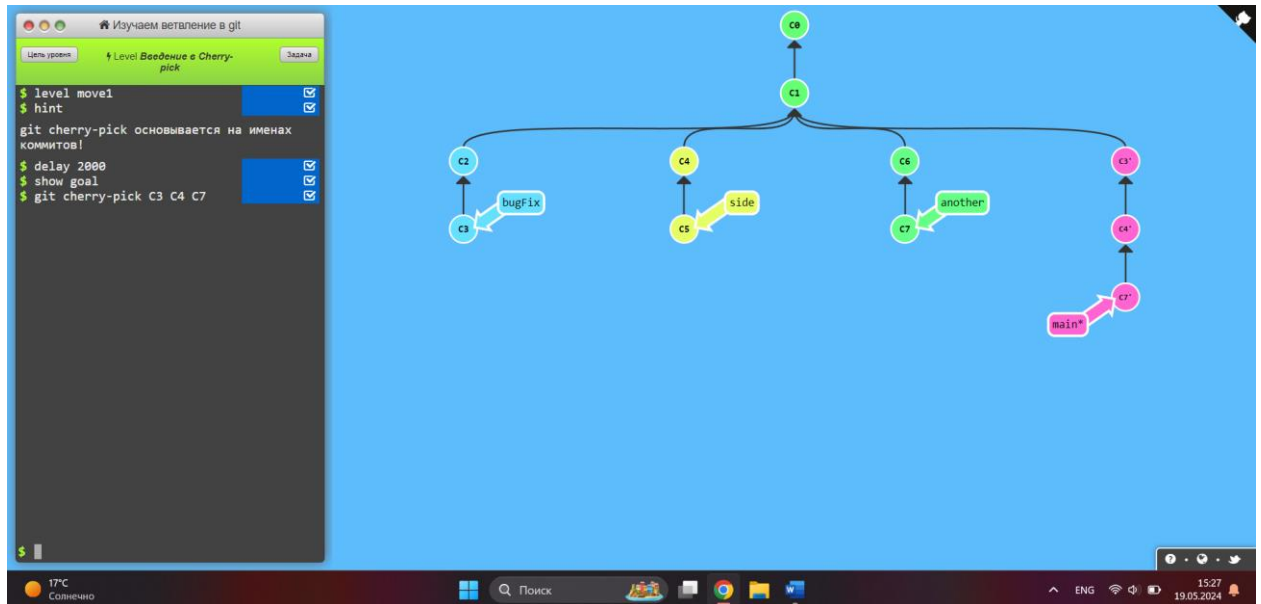
3.1

```
$ level move1
```

```

$ hint
git cherry-pick основывается на именах коммитов!
$ delay 2000
$ show goal
$ git cherry-pick C3 C4 C7

```

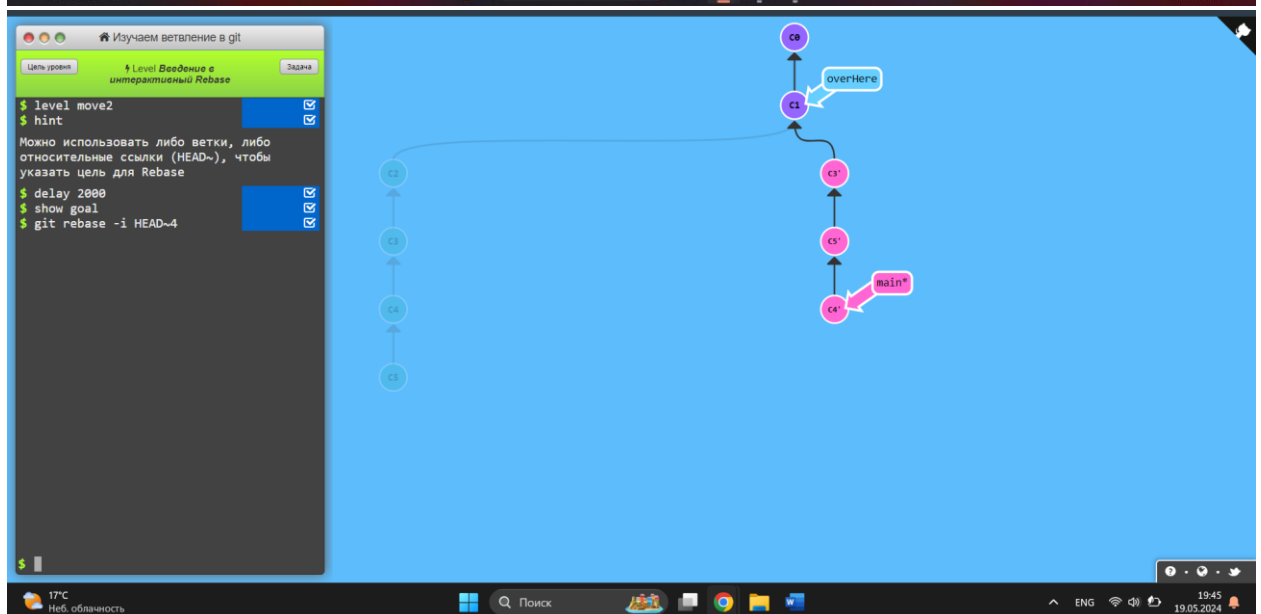
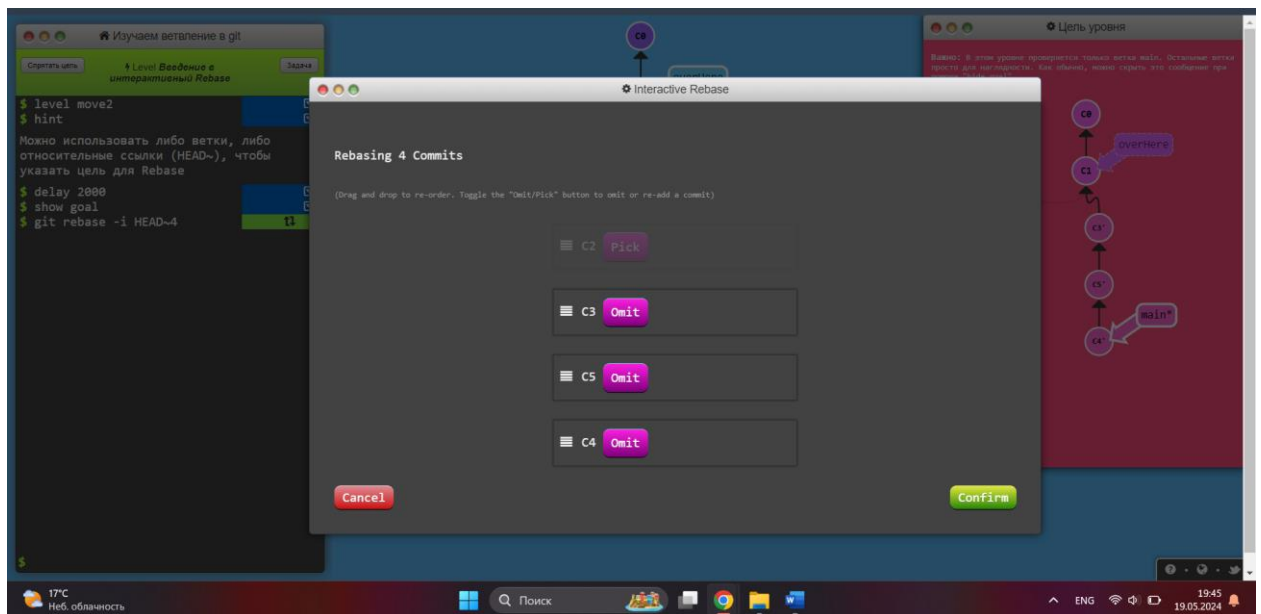


3.2

```

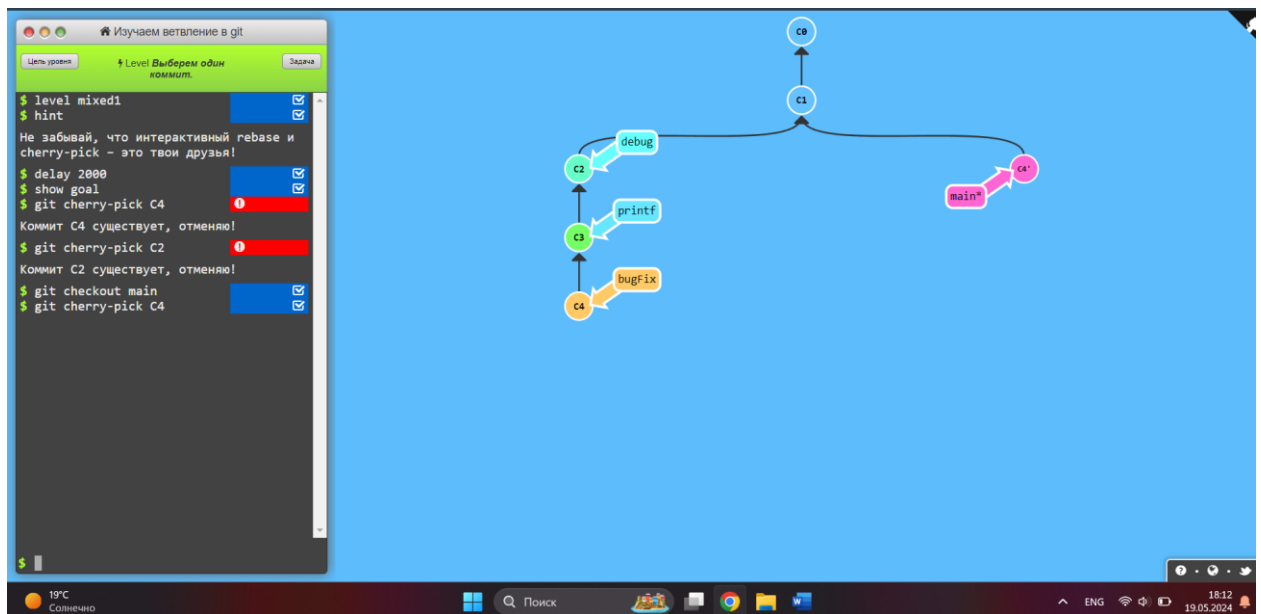
$ level move2
$ hint
Можно использовать либо ветки, либо относительные ссылки
(HEAD~), чтобы указать цель для Rebase
$ delay 2000
$ show goal
$ git rebase -i HEAD~4

```



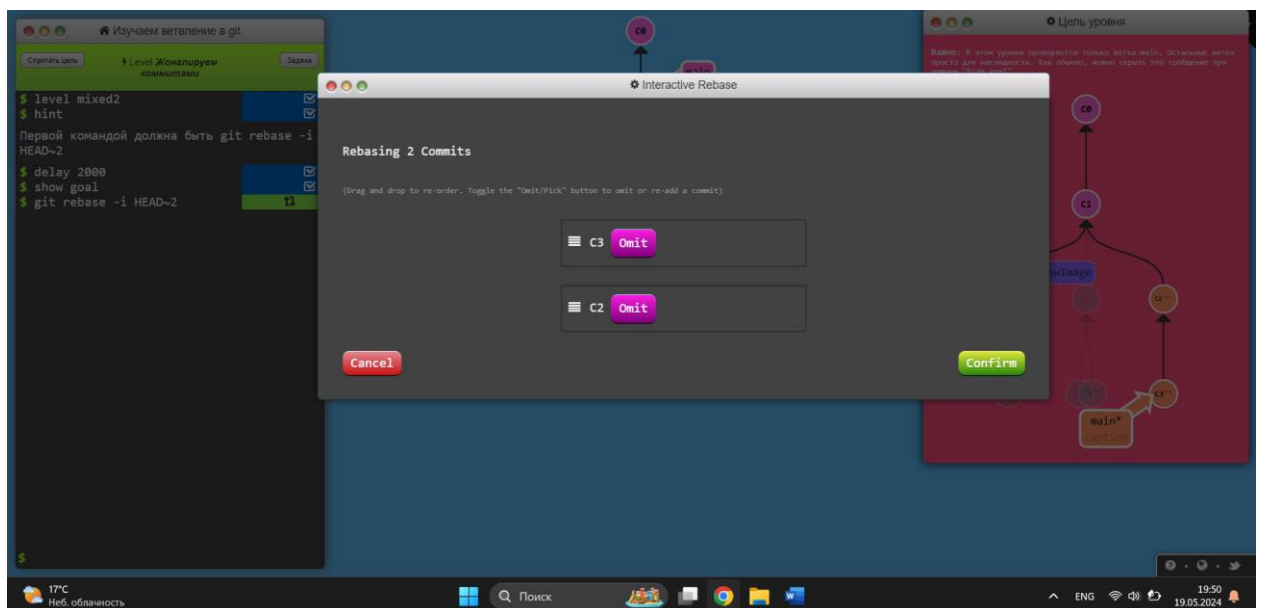
4.1

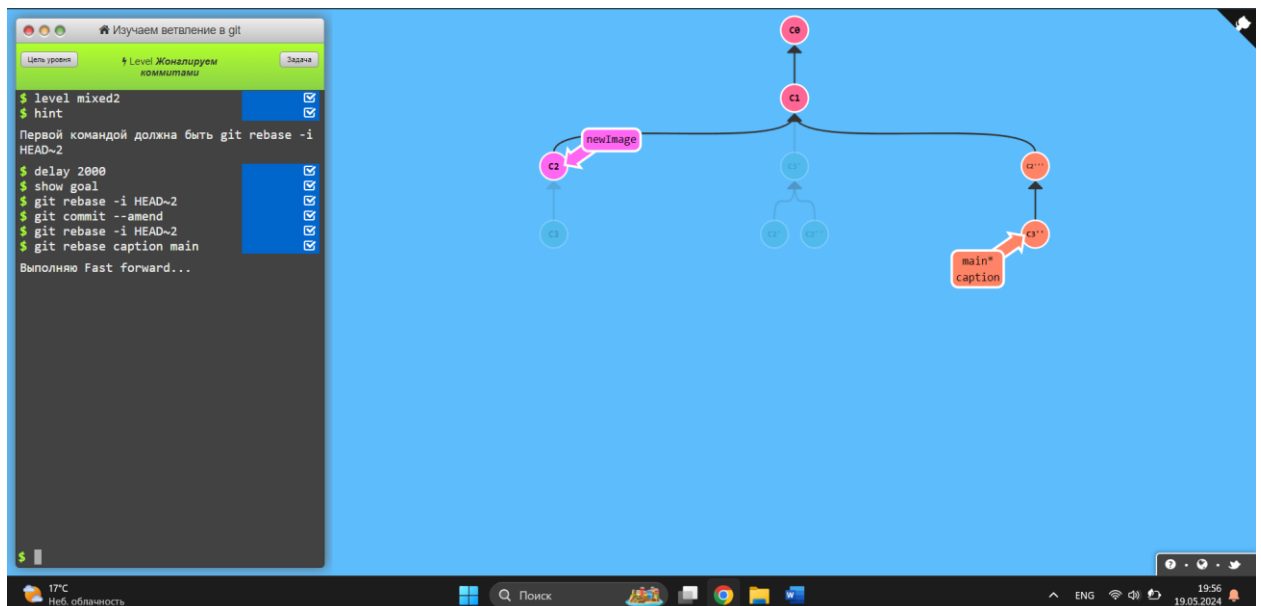
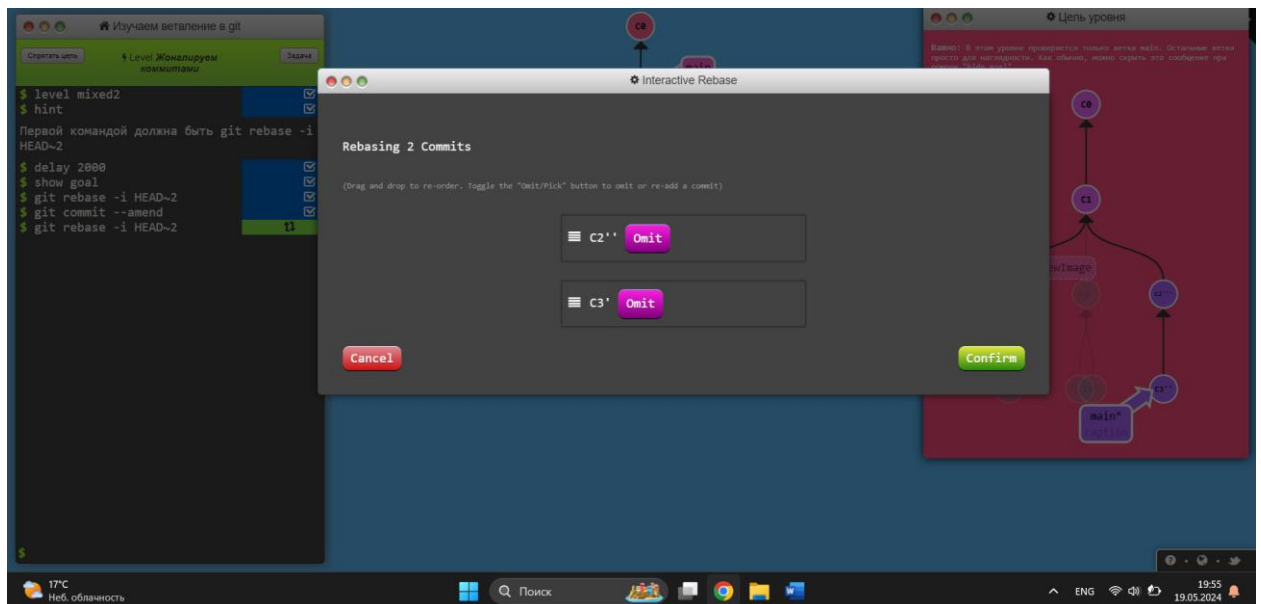
```
$ git checkout main
$ git cherry-pick C4
```

4.2

```
$ level mixed2
$ hint
Первой командой должна быть git rebase -i HEAD~2
$ delay 2000
$ show goal
$ git rebase -i HEAD~2
$ git commit --amend
$ git rebase -i HEAD~2
$ git rebase caption main
Выполняю Fast forward...
```





4.3

```
$ level mixed3
$ hint
Не забудь переместить main на последние изменения.
$ delay 2000
$ show goal
$ git checkout main
$ git cherry-pick C2
$ git checkout C1
$ git cherry-pick C2
$ git cherry-pick C3
$ git branch -f main C3'
```

Исучаем ветвление в git

Цель уровня

Level Жонглируем коммитами 102

Задача

```

$ level mixed3
$ hint
Не забудь переместить main на последние изменения.
$ delay 2000
$ show goal
$ git checkout main
$ git cherry-pick C2
$ git checkout C1
$ git cherry-pick C2
$ git cherry-pick C3
$ git branch -f main C3'
The command "git branc -f main C3'" isn't supported, sorry!
$ git branch -f main C3'

```

```

graph TD
    C0((C0)) --> C1((C1))
    C1 --> C2((C2))
    C2 --> C3((C3))
    C3 --> C2p((C2'))
    C2p --> C3p((C3'))
    C3p --> C0
    newImage[newImage] --> C2
    caption[caption] --> C3
    main[main] --> C3p
    HEAD[HEAD] --> C3p

```

4.4

```

$ level mixed4
$ hint
Можно сделать checkout напрямую на коммит или же на тег
$ delay 2000
$ show goal
$ objective
$ git tag v0 C1
$ git tag v1 C2
$ git checkout C2

```

Исучаем ветвление в git

Цель уровня

Level git tag

Задача

```

$ level mixed4
$ hint
Можно сделать checkout напрямую на коммит или же на тег
$ delay 2000
$ show goal
$ objective
$ git tag v0 C1
$ git tag v1 C2
$ git checkout C2

```

```

graph TD
    C0((C0)) --> C1((C1))
    C1 --> C2((C2))
    C2 --> C3((C3))
    C3 --> C4((C4))
    C4 --> C0
    HEAD[HEAD] --> C2
    side[side] --> C3
    v0[v0] --> C1
    v1[v1] --> C2
    main[main] --> C3

```

4.5

Исучаем ветвление в git

Цель уровня: **Level Git describe**

Задача

```
$ level mixed5
$ hint
Когда закончишь, просто сделай commit
$ delay 2000
$ show goal
$ git checkout C6
$ git commit
Внимание! Репозиторий в состоянии detached HEAD, то есть не находится ни на какой ветке!
$ git branch -f bugFix C7
```

17°C Неб. облачность 20:10 19.05.2024

5.1

```
$ level advanced1
$ hint
Не забудь, что лучше всего сдвинуть мастер в самом конце...
$ delay 2000
$ show goal
$ git rebase main bugFix
$ git rebase bugFix side
$ git rebase side another
$ git rebase another main
Выполняю Fast forward...
```

Исучаем ветвление в git

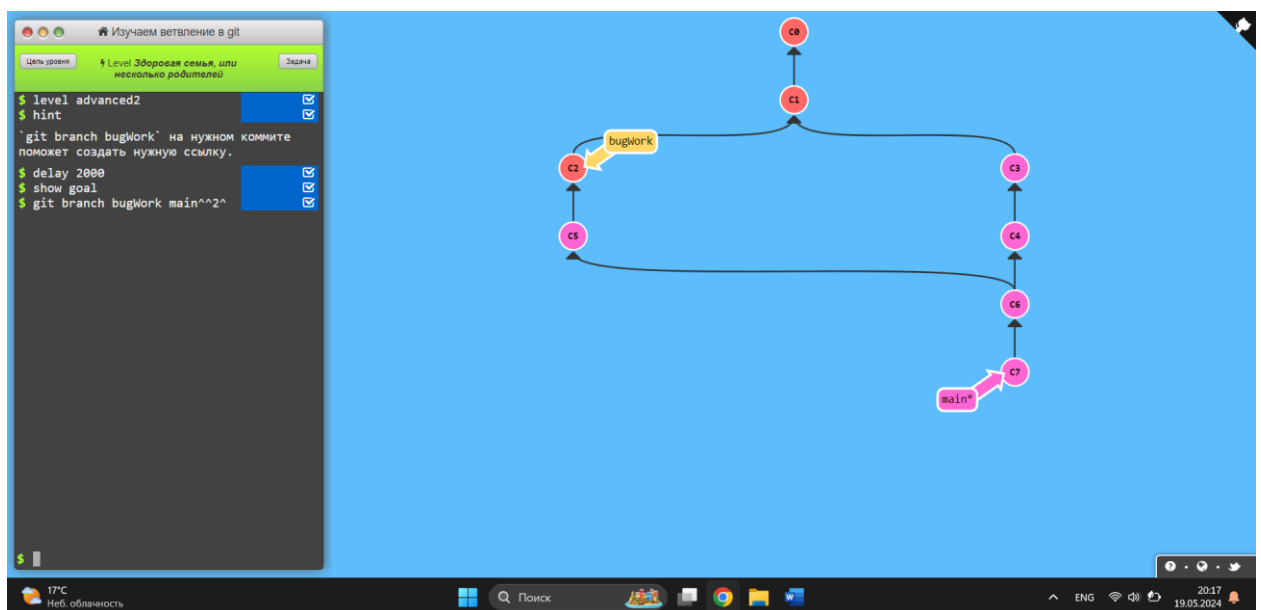
Цель уровня: **Level Rebase over 9000 раз**

Задача

```
$ level advanced1
$ hint
Не забудь, что лучше всего сдвинуть мастер в самом конце...
$ delay 2000
$ show goal
$ git rebase main bugFix
$ git rebase bugFix side
$ git rebase side another
$ git rebase another main
Выполняю Fast forward...
```

5.2

```
$ level advanced2
$ hint
`git branch bugWork` на нужном коммите поможет создать нужную
ссылку.
$ delay 2000
$ show goal
$ git branch bugWork main^^2^
```

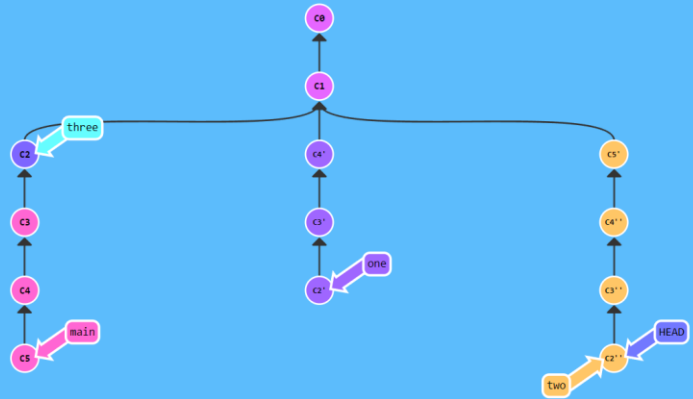


5.3

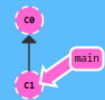
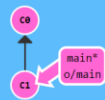
```
$ level advanced3
$ hint
Убедись, что у нас всё по порядку! Сначала ветка `one`, потом
`two` и только потом `three`
$ delay 2000
$ show goal
$ git checkout C1
$ git cherry-pick C4
$ git checkout C4'
$ git cherry-pick C3
$ git cherry-pick C2
$ git checkout C1
```

```
$ level advanced3
$ hint
Убедись, что у нас всё по порядку!
Сначала ветка 'one', потом 'two' и только
потом 'three'

$ delay 2000
$ show goal
$ git checkout C1
$ git cherry-pick C4
$ git checkout C4'
$ git cherry-pick C3
$ git cherry-pick C2
$ git checkout C1
$ git cherry-pick C5
$ git cherry-pick C4
$ git cherry-pick C3
$ git cherry-pick C2
$ git branch -f one C2'
$ git branch -f three C2
$ git branch -f two C2'
```



Удаленные репозитории.

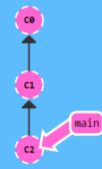
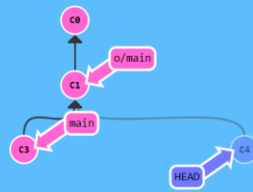


```
Исучаем ветвление в git
Цель уровня: ↑ Level Удалённые ветки
Задача:

$ level remote2
$ hint
Уделяйте внимание очередности -- сперва
commit на main

$ delay 2000
$ show goal
$ git commit
$ git checkout o/main
$ git commit

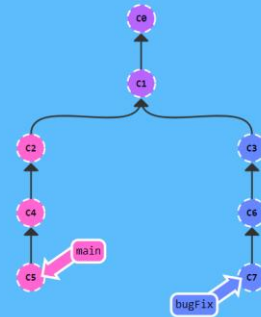
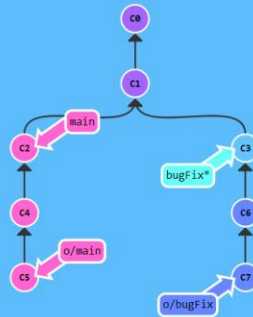
Внимание! Репозиторий в состоянии
detached HEAD, то есть не находится ни на
какой ветке!
```



```
Исучаем ветвление в git
Цель уровня: ↑ Level Git fetch
Задача:

$ level remote3
$ hint
Просто выполните git fetch!

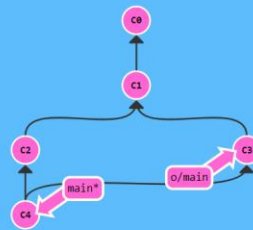
$ delay 2000
$ show goal
$ git fetch
```



```
Исучаем ветвление в git
Цель уровня: ↑ Level Git pull
Задача:

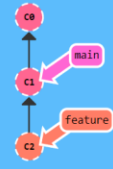
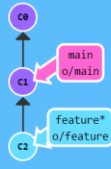
$ level remote4
$ hint
Запустите команду git pull !

$ delay 2000
$ show goal
$ git pull
```



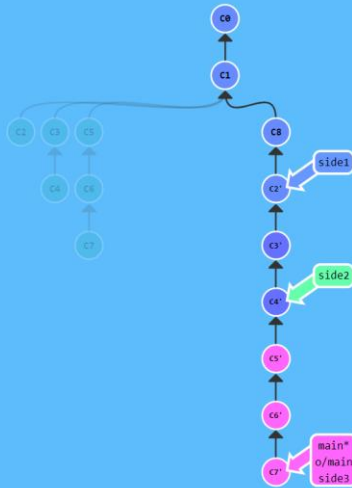

```
Исучаем ветвление в git
Цель уровня: Level Заблокированная ветвь main
Задача:

$ level remote8
$ hint
Создайте новую feature ветвь от main
перед тем, как откатить изменения в main
до состояния o/main.
$ delay 2000
$ show goal
$ git reset --hard o/main
По умолчанию будет выполнен --hard
reset, эту опцию можно опускать!
$ git checkout -b feature C2
$ git push origin feature
local branch "feature" set to track
remote branch "o/feature"
```



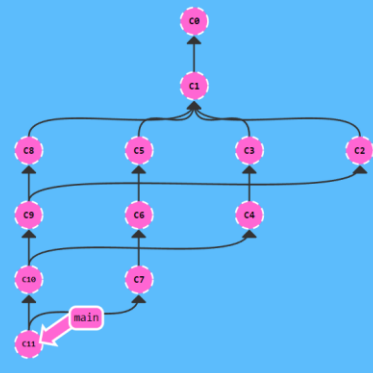
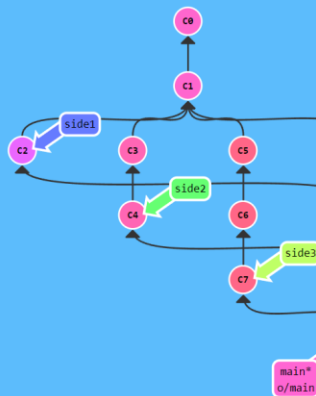
```
Исучаем ветвление в git
Цель уровня: Level Push Мастер!
Задача:

$ level remoteAdvanced1
$ hint
Помни - ты всегда можешь отменить команды
с помощью undo или reset
$ delay 2000
$ show goal
$ git fetch
$ git rebase o/main side1
$ git rebase side1 side2
$ git rebase side2 side3
$ git rebase side3 main
Выполняю Fast forward...
$ git push
```



```
Исучаем ветвление в git
Цель уровня: Level Слияние с удалённым
репозиторием
Задача:

$ level remoteAdvanced2
$ hint
Внимательно посмотрите на цель уровня!
$ delay 2000
$ show goal
$ git checkout main
$ git pull
Выполняю Fast forward...
$ git merge side1
$ git merge side2
$ git merge side3
$ git push
```



Исучаем ветвление в git

Цель уровня

Level 3Слелка за удалённым репозиторием

Задана

\$ level remoteAdvanced3

\$ hint

Помни, есть два способа установить слежку за удалённым репозиторием!

\$ delay 2000

\$ show goal

\$ git checkout -b side o/main

Local branch "side" set to track remote branch "o/main"

\$ git commit

\$ git pull --rebase

\$ git push

Исучаем ветвление в git

Цель уровня

Level 4Аргументы git push

Задана

\$ level remoteAdvanced4

\$ hint

Вы всегда можете ознакомиться с последним слайдом, воспользовавшись "objective".

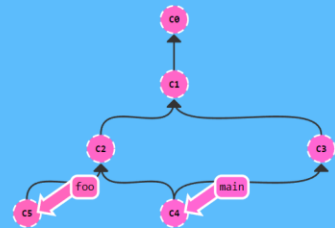
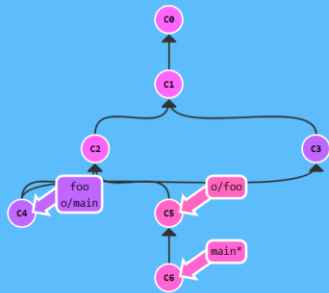
\$ delay 2000

\$ show goal

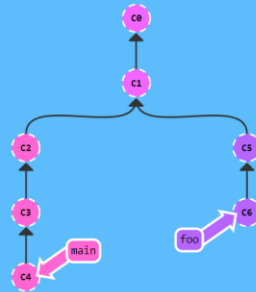
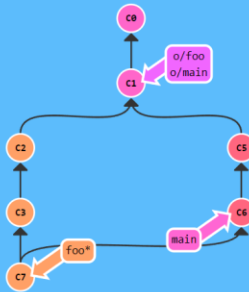
\$ git push origin main

\$ git push origin foo

```
Исучаем ветвление в git
Цель уровня: Level Аргументы для push - расширенная версия!
$ level remoteAdvanced5
$ hint
Помните, Вы всегда можете признать своё поражение, набрав команду "show solution" (показать решение) :P
$ delay 2000
$ show goal
$ git push origin main^:foo
$ git push origin foo:main
```



```
Исучаем ветвление в git
Цель уровня: Level Аргументы для fetch
$ level remoteAdvanced6
$ hint
Обратите внимание на то, как номера коммитов могут меняться! Вы можете прочесть слайды вновь, воспользовавшись командой "help level"
$ delay 2000
$ show goal
$ git fetch origin c3:foo
$ git fetch origin c6:main
$ git checkout foo
$ git merge main
```



Исучаем ветвление в git

Цель уровня

Level Пустой источник

Задача

\$ level remoteAdvanced7

\$ hint

Команда branch недоступна на этом упражнении, пользуйтесь командой fetch!

\$ delay 2000

\$ show goal

\$ git push origin :foo

\$ git fetch origin :bar

✓

✓

✓

✓

✓

✓

Исучаем ветвление в git

Цель уровня

Level Аргументы для pull

Задача

\$ level remoteAdvanced8

\$ hint

Напоминаю, что новые ветки можно создавать и с помощью команд fetch/pull

\$ delay 2000

\$ show goal

\$ git pull origin c3:foo

\$ git pull origin c2:side

✓

✓

✓

✓

✓

✓