Федеральное государственное бюджетное образовательное учреждение высшего образования «Национальный исследовательский университет «МЭИ»

Институт информационных и вычислительных технологий

Кафедра Управления и интеллектуальных технологий

**Отчёт по лабораторной работе № 2**

# По курсу «Разработка ПО систем управления»

# «Система контроля версий Git»

# Выполнил студент группы А-02-19

# Выполнил Ли Егор Андреевич

# Проверили

# Мохов А. С.

# Козлюк Д. А.

Москва 2020

# Цель работы

1. Знать понятия и компоненты систем контроля версий (СКВ), порядок и приемы работы с ними.
2. Уметь участвовать в командной разработке, используя конкретную СКВ — Git, а также популярный хостинг репозитариев — GitHub.

Отчёт Алиса :

Егор@Egor MINGW32 ~/Desktop/lab02

$ mkdir alice

Егор@Egor MINGW32 ~/Desktop/lab02

$ mkdir bob

создаём папки алисы и боба

Егор@Egor MINGW32 ~/Desktop/lab02

$ cd alice

переход в папку Алиса

Егор@Egor MINGW32 ~/Desktop/lab02/alice

$ mkdir project

создали папку проджект в папке алиса

Егор@Egor MINGW32 ~/Desktop/lab02/alice

$ cd project

перешли в папку проджект

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project

$ cd ..

Егор@Egor MINGW32 ~/Desktop/lab02/alice

перешли на уровень выше

Егор@Egor MINGW32 ~/Desktop/lab02/alice

$ cd ..

Егор@Egor MINGW32 ~/Desktop/lab02

$ cd alice

Егор@Egor MINGW32 ~/Desktop/lab02/alice

$ cd project

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project

сначала перешли на уровень выше , а потом вернулись в папку проджект

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git init

Initialized empty Git repository in C:/Users/Егор/Desktop/lab02/alice/project/project/.git/

иницизируем репозитарий в текущем каталоге

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project (master)

$ git config user.name 'alice (LiEA)'

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project (master)

$ git config user.email 'alice@exaample.com'

настройка репозитарий алисы

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git status

On branch master данная фраза ознаает , что мы находимся на ветке мастера

No commits yet данная фраза означает , что нет коммитов

Untracked files:  означает, что Git видит файл, которого не было в предыдущем снимке состояния (коммите)

(use "git add <file>..." to include in what will be committed)

bin/

main.cpp

obj/

project.cbp

nothing added to commit but untracked files present (use "git add" to track) означает , что Ничего не добавлено для фиксации, но есть неотслеживаемые файлы

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git add main.cpp

Добавим файл main.cpp в отслеживаемые (в индекс)

$ git status

On branch master

No commits yet

Changes to be committed:

(use "git rm --cached <file>..." to unstage)

new file: main.cpp

Untracked files:

(use "git add <file>..." to include in what will be committed)

bin/

obj/

project.cbp

все тоже самое , что и в пункте до этого , но мы добавили файл main.cpp в отслеживаемый индекс

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git commit -m 'code: заготовка программы'

[master (root-commit) b7d9eb9] code: заготовка программы

1 file changed, 9 insertions(+)

create mode 100644 main.cpp

коммитим заготовку программы

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git add project.cbp

warning: LF will be replaced by CRLF in project.cbp.

The file will have its original line endings in your working directory

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git status

On branch master

Changes to be committed:

(use "git reset HEAD <file>..." to unstage)

new file: project.cbp

Untracked files:

(use "git add <file>..." to include in what will be committed)

bin/

obj/

добавляем новый файл в индекс и проверяем статус

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ commit -m 'build: добавлен файл проекта'

bash: commit: command not found

коммитим

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git status

On branch master

Changes to be committed:

(use "git reset HEAD <file>..." to unstage)

new file: project.cbp

Changes staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git checkout -- <file>..." to discard changes in working directory)

modified: main.cpp *модифицирован файл main.cpp и не добавл*

Untracked files:

(use "git add <file>..." to include in what will be committed)

bin/

obj/

проверяем статус и попытаемся найти разницу между изменением кода и добавлении

увидили разницу в добавлении в изменении кода , выводятся на экран разный текст

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git add main.cpp

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git commit -m "change code"

[master fb7d5b3] change code

2 files changed, 46 insertions(+), 2 deletions(-)

create mode 100644 project.cbp

из этого видим , что действительно мы поменяли код в main.cpp

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git add -u

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git commit -m"code: изменение кода вывод суммы "

[master aaf7e98] code: изменение кода вывод суммы

1 file changed, 1 insertion(+)

ввели код на вывод суммы и закомитили

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git commit -a -m"code: изменение кода вывод разности"

[master 6ddb108] code: изменение кода вывод разности

1 file changed, 2 insertions(+), 1 deletion(-)

вывели разность и закомитили

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git status

On branch master

Untracked files:

(use "git add <file>..." to include in what will be committed)

.gitignore

obj/

nothing added to commit but untracked files present (use "git add" to track)

заигнорировали bin

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git add .gitignore

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git commit -m "git : игнорирование bin и obj "

[master 6592d51] git : игнорирование bin и obj

1 file changed, 2 insertions(+)

create mode 100644 .gitignore

заигнорировали bin и obj и закоммитили это

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git log --stat

commit 6592d51420f313fcd28facf490f708850ce8faea (HEAD -> master)

Author: alice <alice@example.com>

Date: Fri Mar 27 03:44:28 2020 +0300

git : игнорирование bin и obj *видим что в файл gitignore внесено 2 изменения*

.gitignore | 2 ++

1 file changed, 2 insertions(+)

commit bf494bf054e2c3054b846f5d42d7d90528fb7eb4

Author: alice <alice@example.com>

Date: Fri Mar 27 03:39:19 2020 +0300

code: изменение кода вывод разности

main.cpp | 1 +

1 file changed, 1 insertion(+)

commit 53e674de8e47c6e2a83071ef2903f7881fd1e61d

Author: alice <alice@example.com>

Date: Fri Mar 27 03:37:59 2020 +0300

code: изменение кода вывод суммы

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git log --oneline --decorate

6592d51 (HEAD -> master) git : игнорирование bin и obj

bf494bf code: изменение кода вывод разности

53e674d code: изменение кода вывод суммы

53962f2 code: change code

6b5f28c build: добавлен файл проекта

9c2c128 code: заготовка программы

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git log --oneline --decorate --all --graph

\* 6592d51 (HEAD -> master) git : игнорирование bin и obj

\* bf494bf code: изменение кода вывод разности

\* 53e674d code: изменение кода вывод суммы

\* 53962f2 code: change code

\* 6b5f28c build: добавлен файл проекта

\* 9c2c128 code: заготовка программы

посмотрели журнал репозиатрия с помощью команды git log

рассмотрели его множество операций :

git log --stat показывает файлы, измененные в коммитах;

git log --oneline --decorate показывает коммиты компактно;

git log --oneline --decorate --all --graph делает то же для всех веток.

$ git log --stat

commit 6592d51420f313fcd28facf490f708850ce8faea (HEAD -> master)

Author: alice <alice@example.com>

Date: Fri Mar 27 03:44:28 2020 +0300

git : игнорирование bin и obj

.gitignore | 2 ++

1 file changed, 2 insertions(+)

посмотрим , что показывает git log --stat для последнего коммита

сначала мы видим хэш коммита , (HEAD -> master) означает , что мы находимся на ветке мастера   
автора этого коммита , дату , сколько файлов было изменено и кол-во вставленных строк

$ git log -- main.cpp

commit bf494bf054e2c3054b846f5d42d7d90528fb7eb4

Author: alice <alice@example.com>

Date: Fri Mar 27 03:39:19 2020 +0300

code: изменение кода вывод разности

commit 53e674de8e47c6e2a83071ef2903f7881fd1e61d

Author: alice <alice@example.com>

Date: Fri Mar 27 03:37:59 2020 +0300

code: изменение кода вывод суммы

commit 53962f2cd3ab36e76c8a8b641c9fc2b6467ffead

Author: alice <alice@example.com>

Date: Fri Mar 27 03:36:55 2020 +0300

code: change code

commit 9c2c128fa6f7f2d9917f35960d3c2afed2bf32be

Author: alice <alice@example.com>

Date: Fri Mar 27 03:32:51 2020 +0300

code: заготовка программы

посмотрели коммиты которые затрагивают main.cpp

$ git log --grep "code:"

commit bf494bf054e2c3054b846f5d42d7d90528fb7eb4

Author: alice <alice@example.com>

Date: Fri Mar 27 03:39:19 2020 +0300

code: изменение кода вывод разности

commit 53e674de8e47c6e2a83071ef2903f7881fd1e61d

Author: alice <alice@example.com>

Date: Fri Mar 27 03:37:59 2020 +0300

code: изменение кода вывод суммы

commit 53962f2cd3ab36e76c8a8b641c9fc2b6467ffead

Author: alice <alice@example.com>

Date: Fri Mar 27 03:36:55 2020 +0300

code: change code

commit 9c2c128fa6f7f2d9917f35960d3c2afed2bf32be

Author: alice <alice@example.com>

Date: Fri Mar 27 03:32:51 2020 +0300

code: заготовка программы

показывает коммиты с code: в сообщении.

$ git log --grep "build:"

commit 6b5f28c9649647a0130e581b092115c951d74f35

Author: alice <alice@example.com>

Date: Fri Mar 27 03:34:22 2020 +0300

build: добавлен файл проекта

коммит по теме build

$ git log -- project.cbp

commit 6b5f28c9649647a0130e581b092115c951d74f35

Author: alice <alice@example.com>

Date: Fri Mar 27 03:34:22 2020 +0300

build: добавлен файл проекта

коммит по project.cbp

$ git show 6b5f28c9649647a0130e581b092115c951d74f35

commit 6b5f28c9649647a0130e581b092115c951d74f35

Author: alice <alice@example.com>

Date: Fri Mar 27 03:34:22 2020 +0300

build: добавлен файл проекта

**diff --git a/project.cbp b/project.cbp**

**new file mode 100644**

**index 0000000..e066312**

**--- /dev/null**

**+++ b/project.cbp**

@@ -0,0 +1,43 @@

+<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>

+<CodeBlocks\_project\_file>

посмотрели содержание последнего коммита по теме build

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git show HEAD~1

commit bf494bf054e2c3054b846f5d42d7d90528fb7eb4

Author: alice <alice@example.com>

Date: Fri Mar 27 03:39:19 2020 +0300

code: изменение кода вывод разности

**diff --git a/main.cpp b/main.cpp**

**index 51028c1..8bbdab2 100644**

**--- a/main.cpp**

**+++ b/main.cpp**

@@ -8,4 +8,5 @@ cout << "Enter A and B: ";

int a, b;

cin >> a >> b;

cout << "A + B = " << a + b << '\n' ;

+cout << "A - B = " << a - b << '\n' ;

}

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git show master~1

commit bf494bf054e2c3054b846f5d42d7d90528fb7eb4

Author: alice <alice@example.com>

Date: Fri Mar 27 03:39:19 2020 +0300

code: изменение кода вывод разности

**diff --git a/main.cpp b/main.cpp**

**index 51028c1..8bbdab2 100644**

**--- a/main.cpp**

**+++ b/main.cpp**

@@ -8,4 +8,5 @@ cout << "Enter A and B: ";

int a, b;

cin >> a >> b;

cout << "A + B = " << a + b << '\n' ;

+cout << "A - B = " << a - b << '\n' ;

}

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git show bf494bf054e2c3054b846f5d42d7d90528fb7eb4

commit bf494bf054e2c3054b846f5d42d7d90528fb7eb4

Author: alice <alice@example.com>

Date: Fri Mar 27 03:39:19 2020 +0300

code: изменение кода вывод разности

**diff --git a/main.cpp b/main.cpp**

**index 51028c1..8bbdab2 100644**

**--- a/main.cpp**

**+++ b/main.cpp**

@@ -8,4 +8,5 @@ cout << "Enter A and B: ";

int a, b;

cin >> a >> b;

cout << "A + B = " << a + b << '\n' ;

+cout << "A - B = " << a - b << '\n' ;

}

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$

посмотрели предпоследний коммит тремя способами

+cout << "A - B = " << a - b << '\n' - обозначает добавление добавление 1-ой новой строки

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git diff

**diff --git a/main.cpp b/main.cpp**

**index 8bbdab2..095c47c 100644**

**--- a/main.cpp**

**+++ b/main.cpp**

@@ -9,4 +9,5 @@ int a, b;

cin >> a >> b;

cout << "A + B = " << a + b << '\n' ;

cout << "A - B = " << a - b << '\n' ;

+cout << "A \* B + " << a \* b << '\n' ;

}

пояснение :

**diff --git a/main.cpp b/main.cpp**

- Это заголовок" git diff " в виде **diff --git a/main.cpp b/main.cpp**

Элемент **--git** означает, что diff находится в" git " diff формат.

Имена файлов  **a/**и**b/** совпадают так как не задействовано копирование

--git означает, что diff находится в формате "git" diff.

вторая строка показывает индекс файла расскажите нам о режиме данного файла **100644** означает, что это обычный файл, а не например символьная ссылка, и что он не имеет исполняемого бита разрешения

в третьей строке идет двухстрочный унифицированный заголовок diff й унифицированный заголовок diff

в четвертой строке идёт один или несколько кусков различий; каждый кусок показывает одну область, где файлы отличаются.

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git diff HEAD~2

**diff --git a/.gitignore b/.gitignore**

**new file mode 100644**

**index 0000000..4c7473d**

**--- /dev/null**

**+++ b/.gitignore**

@@ -0,0 +1,2 @@

+/bin

+/obj

**diff --git a/main.cpp b/main.cpp**

**index 51028c1..095c47c 100644**

**--- a/main.cpp**

**+++ b/main.cpp**

@@ -8,4 +8,6 @@ cout << "Enter A and B: ";

int a, b;

cin >> a >> b;

cout << "A + B = " << a + b << '\n' ;

+cout << "A - B = " << a - b << '\n' ;

+cout << "A \* B + " << a \* b << '\n' ;

}

Первый аргумент команды git diff включает показ изменений от указанного коммита до последнего, включая изменения в рабочей копии

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git diff HEAD~2 HEAD

**diff --git a/.gitignore b/.gitignore**

**new file mode 100644**

**index 0000000..4c7473d**

**--- /dev/null**

**+++ b/.gitignore**

@@ -0,0 +1,2 @@

+/bin

+/obj

**diff --git a/main.cpp b/main.cpp**

**index 51028c1..8bbdab2 100644**

**--- a/main.cpp**

**+++ b/main.cpp**

@@ -8,4 +8,5 @@ cout << "Enter A and B: ";

int a, b;

cin >> a >> b;

cout << "A + B = " << a + b << '\n' ;

+cout << "A - B = " << a - b << '\n' ;

}

С двумя аргументами команда показывает разницу между указанными коммитами, например, так можно исключить изменения в рабочей копии из вывода предыдущей команды

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git diff HEAD~3 HEAD~1

**diff --git a/main.cpp b/main.cpp**

**index b903969..8bbdab2 100644**

**--- a/main.cpp**

**+++ b/main.cpp**

@@ -7,4 +7,6 @@ int main()

cout << "Enter A and B: ";

int a, b;

cin >> a >> b;

+cout << "A + B = " << a + b << '\n' ;

+cout << "A - B = " << a - b << '\n' ;

}

изменения между самым первым коммитом и коммитом, добавляющим вывод разности.

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git add main.cpp

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git commit -m "code: вывод произведения"

[master 4e30cad] code: вывод произведения

1 file changed, 1 insertion(+)

закомитили вывыод произведениия

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git reset --hard HEAD~1

HEAD is now at 6592d51 git : игнорирование bin и obj

откатывание комита \

Здесь HEAD~1 указывает на коммит, к которому нужно откатить состояние рабочей копии, а ключ --hard означает, что нужно привести рабочую копию точно к состоянию выбранного коммита

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git checkout HEAD -- main.cpp

Уберем изменения в main.cpp другим способом — откатив этот файл к состоянию в последнем коммите

таким образом , мы откатили 2 последних файла

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git remote add origin https://github.com/Eroproralee/cs-lab03.git

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git push -u origin master

Enumerating objects: 18, done.

Counting objects: 100% (18/18), done.

Delta compression using up to 4 threads

Compressing objects: 100% (16/16), done.

Writing objects: 100% (18/18), 2.20 KiB | 204.00 KiB/s, done.

Total 18 (delta 2), reused 0 (delta 0)

remote: Resolving deltas: 100% (2/2), done.

To https://github.com/Eroproralee/cs-lab03.git

\* [new branch] master -> master

Branch 'master' set up to track remote branch 'master' from 'origin'.

отпарвляем проект на git.hub

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git fetch

remote: Enumerating objects: 5, done.

remote: Counting objects: 100% (5/5), done.

remote: Compressing objects: 100% (2/2), done.

remote: Total 3 (delta 1), reused 3 (delta 1), pack-reused 0

Unpacking objects: 100% (3/3), done.

From https://github.com/Eroproralee/cs-lab03

0279feb..6c71ae8 master -> origin/master

выполняем загрузку изменений с машины боба

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git log --oneline --decorate --all --graph

\* 6c71ae8 (origin/master) code:вывод произведения

\* 0279feb (HEAD -> master) git: игнорирование obj

\* dfeeac6 code: вывод разности

\* 085e4da code: вывод суммы

\* b29ce31 code: ввод чисел

\* c62c715 build: добавлен файл проекта

\* 4bf7067 code: загатовка программы

проверяем все ветки

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git pull --ff-only

Updating 0279feb..6c71ae8

Fast-forward

main.cpp | 1 +

1 file changed, 1 insertion(+)

Продвигаем ветку master к скачанной версии

Убеждаемся, что рабочая копия проекта «у Алисы» соответствует версии «у Боба»

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git add main.cpp

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git commit -m 'code:вывод деления'

[master 2bfaed0] code:вывод деления

1 file changed, 1 insertion(+)

Добавляем деление и коммитим

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git push

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Delta compression using up to 4 threads

Compressing objects: 100% (3/3), done.

Writing objects: 100% (3/3), 392 bytes | 196.00 KiB/s, done.

Total 3 (delta 1), reused 0 (delta 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/Eroproralee/cs-lab002.git

f9eb5df..2bfaed0 master -> master

отправляет на git.hub

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git add main.cpp

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git commit -m 'code:вывод максимума'

[master 9937d27] code:вывод максимума

1 file changed, 8 insertions(+)

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git push

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Delta compression using up to 4 threads

Compressing objects: 100% (3/3), done.

Writing objects: 100% (3/3), 407 bytes | 203.00 KiB/s, done.

Total 3 (delta 1), reused 0 (delta 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/Eroproralee/cs-lab002.git

2bfaed0..9937d27 master -> master

вывод максимума , коммитим и отправляем на git.hub

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git branch double

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git checkout double

Switched to branch 'double'

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (double)

создаём ветку дабл и переключаемся на нее , видим соответсвенное изменение

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (double)

$ git commit -m 'code:меняем тип чисел a b '

[double 4107f3c] code:меняем тип чисел a b

1 file changed, 1 insertion(+), 1 deletion(-)

меняем тип чисел и коммитим

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (double)

$ git checkout master

Switched to branch 'master'

переключаемся на ветку master

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git push

Enumerating objects: 13, done.

Counting objects: 100% (13/13), done.

Delta compression using up to 4 threads

Compressing objects: 100% (9/9), done.

Writing objects: 100% (9/9), 1.05 KiB | 215.00 KiB/s, done.

Total 9 (delta 3), reused 0 (delta 0)

remote: Resolving deltas: 100% (3/3), completed with 1 local object.

To https://github.com/Eroproralee/cs-lab002.git

a3c2d93..7fd8009 master -> master

отпарвляем коммти на git.hub

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git pull

remote: Enumerating objects: 5, done.

remote: Counting objects: 100% (5/5), done.

remote: Compressing objects: 100% (2/2), done.

remote: Total 3 (delta 1), reused 3 (delta 1), pack-reused 0

Unpacking objects: 100% (3/3), done.

From https://github.com/Eroproralee/cs-lab002

9937d27..a3c2d93 master -> origin/master

Updating 9937d27..a3c2d93

Fast-forward

main.cpp | 9 ++++++++-

1 file changed, 8 insertions(+), 1 deletion(-)

синхронизируем ветку мастер с github

commit a3c2d933b88841df086d6c41181f162aff561d6f (HEAD -> master, origin/master)

Author: bob (LiEA) <bob@example.com>

Date: Fri Mar 27 13:01:41 2020 +0300

code:вывод минимума

commit 9937d27d3ee318637f085fa9e276148adc8bb1de

Author: alice <alice@example.com>

Date: Fri Mar 27 12:58:51 2020 +0300

code:вывод максимума

commit 2bfaed0f5ed88c457d7800f96c45c8bd67c3a06f

Author: alice <alice@example.com>

Date: Fri Mar 27 12:49:42 2020 +0300

code:вывод деления

commit f9eb5dff23a0212936749db17a553dc48927b3ff

Author: bob (LiEA) <bob@example.com>

Date: Fri Mar 27 12:45:46 2020 +0300

code:вывод произведения

commit a1bf04e58244d92d7cb3b55a86066995137b47f0

Author: alice <alice@example.com>

Date: Fri Mar 27 12:29:43 2020 +0300

git:игнорирование bin и obj

commit d80ca0336ad85fa722447779e694463b1a1d1f97

Author: alice <alice@example.com>

Date: Fri Mar 27 12:27:48 2020 +0300

code:вывод разности

история всех веток

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git merge double

$ $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$git merge double

Auto-merging main.cpp

Merge made by the 'recursive' strategy.

main.cpp | 2 +-

1 file changed, 1 insertion(+), 1 deletion(-)

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ git log

commit 7b27a1c1c367b2b027b5e0f62dd643910572391f (HEAD -> master)

Merge: a3c2d93 4107f3c

Author: alice <alice@example.com>

Date: Fri Mar 27 13:58:37 2020 +0300

branch 'double'

слияне ветки double и master qa!

commit 4107f3cf519cfb1477cefe94c99fd0b47a18e00e (double)

Author: alice <alice@example.com>

Date: Fri Mar 27 13:39:51 2020 +0300

code:меняем тип чисел a b

commit a3c2d933b88841df086d6c41181f162aff561d6f (origin/master)

Author: bob (LiEA) <bob@example.com>

Date: Fri Mar 27 13:01:41 2020 +0300

code:вывод минимума

commit 9937d27d3ee318637f085fa9e276148adc8bb1de

Author: alice <alice@example.com>

Date: Fri Mar 27 12:58:51 2020 +0300

code:вывод максимума

commit 2bfaed0f5ed88c457d7800f96c45c8bd67c3a06f

Author: alice <alice@example.com>

Date: Fri Mar 27 12:49:42 2020 +0300

code:вывод деления

commit f9eb5dff23a0212936749db17a553dc48927b3ff

Author: bob (LiEA) <bob@example.com>

Date: Fri Mar 27 12:45:46 2020 +0300

code:вывод произведения

commit a1bf04e58244d92d7cb3b55a86066995137b47f0

Author: alice <alice@example.com>

Date: Fri Mar 27 12:29:43 2020 +0300

git:игнорирование bin и obj

commit d80ca0336ad85fa722447779e694463b1a1d1f97

Author: alice <alice@example.com>

Date: Fri Mar 27 12:27:48 2020 +0300

code:вывод разности

commit c46e9304e3ce63baea52ee5c16dd8deff85554ae

Author: alice <alice@example.com>

Date: Fri Mar 27 12:27:06 2020 +0

история репозитария

Егор@Egor MINGW32 ~/Desktop/lab02/alice/project/project (master)

$ EDITOR=nano git merge double

Already up to date.

указываем другой оператор , чтобы писать длинные сообщения , но не использовать vim

Отчет Боб:

Егор@Egor MINGW32 ~/Desktop/lab02/bob

$ git clone https://github.com/Eroproralee/cs-lab03.git project

Cloning into 'project'...

remote: Enumerating objects: 18, done.

remote: Counting objects: 100% (18/18), done.

remote: Compressing objects: 100% (14/14), done.

remote: Total 18 (delta 2), reused 18 (delta 2), pack-reused 0

Unpacking objects: 100% (18/18), done.

Открываем новый терминал Git Bash в каталоге bob.

Егор@Egor MINGW32 ~/Desktop/lab02/bob

$ cd project

заходим в папку пролджект

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git config user.name 'bob(LiEA)'

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git config user.email 'bob@example.com'

настраиваем Git (git config)

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git add main.cpp

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git commit -m 'code:вывод произведения'

[master 6c71ae8] code:вывод произведения

1 file changed, 1 insertion(+)

вводим произведение и коммитим

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git push

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Delta compression using up to 4 threads

Compressing objects: 100% (3/3), done.

Writing objects: 100% (3/3), 408 bytes | 204.00 KiB/s, done.

Total 3 (delta 1), reused 0 (delta 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/Eroproralee/cs-lab03.git

0279feb..6c71ae8 master -> master

отправляем на git.hub

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git fetch

remote: Enumerating objects: 5, done.

remote: Counting objects: 100% (5/5), done.

remote: Compressing objects: 100% (2/2), done.

remote: Total 3 (delta 1), reused 3 (delta 1), pack-reused 0

Unpacking objects: 100% (3/3), done.

From https://github.com/Eroproralee/cs-lab002

f9eb5df..2bfaed0 master -> origin/master

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git log --oneline --decorate --all --graph

\* 2bfaed0 (origin/master, origin/HEAD) code:вывод деления

\* f9eb5df (HEAD -> master) code:вывод произведения

\* a1bf04e git:игнорирование bin и obj

\* d80ca03 code:вывод разности

\* c46e930 code:ввод суммы

\* d95d88d code:change code

\* fe14305 build: Добавлен файл проекта

\* 9f83b6c code: заготовка программы

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git pull --ff-only

Updating f9eb5df..2bfaed0

Fast-forward

main.cpp | 1 +

1 file changed, 1 insertion(+)

получаем новую версию с делением с машины Алисы на машину Боба

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git add main.cpp

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git commit -m 'code:вывод минимума'

[master 565d978] code:вывод минимума

1 file changed, 8 insertions(+)

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git push

To https://github.com/Eroproralee/cs-lab002.git

! [rejected] master -> master (fetch first)

error: failed to push some refs to 'https://github.com/Eroproralee/cs-lab002.git'

hint: Updates were rejected because the remote contains work that you do

hint: not have locally. This is usually caused by another repository pushing

hint: to the same ref. You may want to first integrate the remote changes

hint: (e.g., 'git pull ...') before pushing again.

hint: See the 'Note about fast-forwards' in 'git push --help' for details.

ищем минимум , комитим и пытаемся отправить на git.hub

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git fetch

remote: Enumerating objects: 5, done.

remote: Counting objects: 100% (5/5), done.

remote: Compressing objects: 100% (2/2), done.

remote: Total 3 (delta 1), reused 3 (delta 1), pack-reused 0

Unpacking objects: 100% (3/3), done.

From https://github.com/Eroproralee/cs-lab002

2bfaed0..9937d27 master -> origin/master

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git log

commit 565d9782b6117bd98307435d83dce0df2f013381 (HEAD -> master)

Author: bob (LiEA) <bob@example.com>

Date: Fri Mar 27 13:01:41 2020 +0300

code:вывод минимума

commit 2bfaed0f5ed88c457d7800f96c45c8bd67c3a06f

Author: alice <alice@example.com>

Date: Fri Mar 27 12:49:42 2020 +0300

code:вывод деления

commit f9eb5dff23a0212936749db17a553dc48927b3ff

Author: bob (LiEA) <bob@example.com>

Date: Fri Mar 27 12:45:46 2020 +0300

code:вывод произведения

commit a1bf04e58244d92d7cb3b55a86066995137b47f0

Author: alice <alice@example.com>

Date: Fri Mar 27 12:29:43 2020 +0300

git:игнорирование bin и obj

загружаем коммиты из удаленного хранилища и отобразите историю всех веток

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git rebase origin/master

First, rewinding head to replay your work on top of it...

Applying: code:вывод минимума

Using index info to reconstruct a base tree...

M main.cpp

Falling back to patching base and 3-way merge...

Auto-merging main.cpp

CONFLICT (content): Merge conflict in main.cpp

error: Failed to merge in the changes.

hint: Use 'git am --show-current-patch' to see the failed patch

Patch failed at 0001 code:вывод минимума

Resolve all conflicts manually, mark them as resolved with

"git add/rm <conflicted\_files>", then run "git rebase --continue".

You can instead skip this commit: run "git rebase --skip".

To abort and get back to the state before "git rebase", run "git rebase --abort".

перемещаем коммит боба поверх коммита алисы , однако возникает ошибка , которая означает ,что возникают конфликты , которые надо удалить вручную

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git add main.cpp

Егор@Egor MINGW32 ~/Desktop/lab02/bob/project (master)

$ git rebase --countine

error: unknown option `countine'

usage: git rebase [-i] [options] [--exec <cmd>] [--onto <newbase>] [<upstream>] [<branch>]

or: git rebase [-i] [options] [--exec <cmd>] [--onto <newbase>] --root [<branch>]

or: git rebase --continue | --abort | --skip | --edit-todo

--onto <revision> rebase onto given branch instead of upstream

--no-verify allow pre-rebase hook to run

-q, --quiet be quiet. implies --no-stat

-v, --verbose display a diffstat of what changed upstream

-n, --no-stat do not show diffstat of what changed upstream

--signoff add a Signed-off-by: line to each commit

--ignore-whitespace passed to 'git am'

--committer-date-is-author-date

passed to 'git am'

--ignore-date passed to 'git am'

-C <n> passed to 'git apply'

--whitespace <action>

passed to 'git apply'

-f, --force-rebase cherry-pick all commits, even if unchanged

--no-ff cherry-pick all commits, even if unchanged

--continue continue

--skip skip current patch and continue

--abort abort and check out the original branch

--quit abort but keep HEAD where it is

--edit-todo edit the todo list during an interactive rebase

--show-current-patch show the patch file being applied or merged

-m, --merge use merging strategies to rebase

-i, --interactive let the user edit the list of commits to rebase

-p, --preserve-merges

try to recreate merges instead of ignoring them

--rerere-autoupdate allow rerere to update index with resolved conflict

-k, --keep-empty preserve empty commits during rebase

--autosquash move commits that begin with squash!/fixup! under -i

-S, --gpg-sign[=<key-id>]

GPG-sign commits

--autostash automatically stash/stash pop before and after

-x, --exec <exec> add exec lines after each commit of the editable list

--allow-empty-message

allow rebasing commits with empty messages

-r, --rebase-merges[=<mode>]

try to rebase merges instead of skipping them

--fork-point use 'merge-base --fork-point' to refine upstream

-s, --strategy <strategy>

use the given merge strategy

-X, --strategy-option <option>

pass the argument through to the merge strategy

--root rebase all reachable commits up to the root(s)

добавляем файл в индекс и продолжаем прерванную операцию rebase