Федеральное государственное автономное образовательное учреждение высшего образования **«Национальный исследовательский университет ИТМО»**

Факультет Программной Инженерии и Компьютерной Техники

**Лабораторная работа №2**

по дисциплине **«Основы программной инженерии»**

Вариант: **9130**

**Преподаватель**:   
Вадим Миху

**Выполнил**: Алферов Г.А.

**Группа**: Р3207

Санкт-Петербург, 2025

# Задание

Изображение выглядит как текст, снимок экрана, Шрифт, число

Контент, сгенерированный ИИ, может содержать ошибки.

# Реализация с использованием Git

rm -rf .git

rm -f .gitignore

rm -rf src

git init

echo "git init"

# Пользователи

git config user.name "red"

git config user.email "red@mail.com"

echo "Красный создан"

git checkout -b branch1 # 0-7-11-14

echo "commits" > .gitignore

echo "git.sh" >> .gitignore

git add .gitignore

echo ".gitignore создан"

# r0 red

unzip -o ../commits/commit0.zip -d src

git add .

git commit -m "Initial commit r0"

echo "r0 создана red"

git checkout -b branch2 # 1-5-6-13

# r1 blue

unzip -o ../commits/commit1.zip -d src

git add .

git commit --author="blue <blue@mail.com>" -m "Revision r1"

echo "r1 создана blue"

git checkout -b branch3 # 2-3-4-8-9-10-12

# r2 blue

unzip -o ../commits/commit2.zip -d src

git add .

git commit --author="blue <blue@mail.com>" -m "Revision r2"

echo "r2 создана blue"

# r3 blue

unzip -o ../commits/commit3.zip -d src

git add .

git commit --author="blue <blue@mail.com>" -m "Revision r3"

echo "r3 создана blue"

# r4 blue

unzip -o ../commits/commit4.zip -d src

git add .

git commit --author="blue <blue@mail.com>" -m "Revision r4"

echo "r4 создана blue"

git checkout branch2 # 1-5-6-13

# r5 blue

unzip -o ../commits/commit5.zip -d src

git add .

git commit --author="blue <blue@mail.com>" -m "Revision r5"

echo "r5 создана blue"

# r6 blue

unzip -o ../commits/commit6.zip -d src

git add .

git commit --author="blue <blue@mail.com>" -m "Revision r6"

echo "r6 создана blue"

git checkout branch1 # 0-7-11-14

#r7 red

unzip -o ../commits/commit7.zip -d src

git add .

git commit -m "Revision r7"

echo "r7 создана red"

git checkout branch3 # 2-3-4-8-9-10-12

# r8 blue

unzip -o ../commits/commit8.zip -d src

git add .

git commit --author="blue <blue@mail.com>" -m "Revision r8"

echo "r8 создана blue"

# r9 blue

unzip -o ../commits/commit9.zip -d src

git add .

git commit --author="blue <blue@mail.com>" -m "Revision r9"

echo "r9 создана blue"

# r10 blue

unzip -o ../commits/commit10.zip -d src

git add .

git commit --author="blue <blue@mail.com>" -m "Revision r10"

echo "r10 создана blue"

git checkout branch1 # 0-7-11-14

#r11 red

unzip -o ../commits/commit11.zip -d src

git add .

git commit -m "Revision r11"

echo "r11 создана red"

git checkout branch3 # 2-3-4-8-9-10-12

# r12 blue

unzip -o ../commits/commit12.zip -d src

git add .

git commit --author="blue <blue@mail.com>" -m "Revision r12"

echo "r12 создана blue"

git checkout branch2 # 1-5-6-13

git merge --no-commit branch3 -Xtheirs

##

## Xtheirs входящие изменения применяются то есть из 3 (Xours наоборот)

##

git add .

echo "Слияние r6 и r12"

# r13 blue

unzip -o ../commits/commit13.zip -d src

git add .

git commit --author="blue <blue@mail.com>" -m "Revision r13"

echo "r13 создана blue"

git checkout branch1 # 0-7-11-14

git merge --no-commit branch2 -Xtheirs

##

## Xtheirs входящие изменения применяются то есть из 2

##

git add .

echo "Слияние r11 и r13"

#r14 red

unzip -o ../commits/commit14.zip -d src

git add .

git commit -m "Revision r14"

echo "r14 создана red"

git log --graph --abbrev-commit --decorate --format=format:'%C(bold blue)%h%C(reset) - %C(bold green)(%ar)%C(reset) %C(white)%s%C(reset) %C(dim white)- %an%C(reset)%C(auto)%d%C(reset)' --all

# Реализация с использованием SVN

rm -rf svn\_repo

rm -rf wc

svnadmin create svn\_repo

REPO\_URL="file:///C:/Users/gleb/OPI2/svn\_repo"

echo "$REPO\_URL"

cd svn\_repo

svn mkdir -m "project structure" "$REPO\_URL/trunk" "$REPO\_URL/branches"

cd ..

svn checkout "$REPO\_URL/trunk"/ wc # 0-7-11-14

cd wc

# r0 red

unzip -o ../commits/commit0.zip -d .

svn add \* --force

svn commit -m "Initial commit (r0)" --username=red

echo "Revision 0 red"

svn copy "$REPO\_URL/trunk" "$REPO\_URL/branches/branch2" -m "Creating branch2" --username=blue # 1-5-6-13

svn switch "$REPO\_URL/branches/branch2"

# r1 blue

unzip -o ../commits/commit1.zip -d .

svn add \* --force

svn commit -m "r1 blue" --username=blue

echo "Revision 1 blue"

svn copy "$REPO\_URL/trunk" "$REPO\_URL/branches/branch3" -m "Creating branch3" # 2-3-4-8-9-10-12

svn switch "$REPO\_URL/branches/branch3"

# r2 blue

unzip -o ../commits/commit2.zip -d .

svn add \* --force

svn commit -m "r2 blue" --username=blue

echo "Revision 2 blue"

# r3 blue

unzip -o ../commits/commit3.zip -d .

svn add \* --force

svn commit -m "r3 blue" --username=blue

echo "Revision 3 blue"

# r4 blue

unzip -o ../commits/commit4.zip -d .

svn add \* --force

svn commit -m "r4 blue" --username=blue

echo "Revision 4 blue"

svn switch "$REPO\_URL/branches/branch2"

# r5 blue

unzip -o ../commits/commit5.zip -d .

svn add \* --force

svn commit -m "r5 blue" --username=blue

echo "Revision 5 blue"

# r6 blue

unzip -o ../commits/commit6.zip -d .

svn add \* --force

svn commit -m "r6 blue" --username=blue

echo "Revision 6 blue"

svn switch "$REPO\_URL/trunk"

# r7 red

unzip -o ../commits/commit7.zip -d .

svn add \* --force

svn commit -m "r7 red" --username=red

echo "Revision 7 red"

svn switch "$REPO\_URL/branches/branch3"

# r8 blue

unzip -o ../commits/commit8.zip -d .

svn add \* --force

svn commit -m "r8 blue" --username=blue

echo "Revision 8 blue"

# r9 blue

unzip -o ../commits/commit9.zip -d .

svn add \* --force

svn commit -m "r9 blue" --username=blue

echo "Revision 9 blue"

# r10 blue

unzip -o ../commits/commit10.zip -d .

svn add \* --force

svn commit -m "r10 blue" --username=blue

echo "Revision 10 blue"

svn switch "$REPO\_URL/trunk"

# r11 red

unzip -o ../commits/commit11.zip -d .

svn add \* --force

svn commit -m "r11 red" --username=red

echo "Revision 11 red"

svn switch "$REPO\_URL/branches/branch3"

# r12 blue

unzip -o ../commits/commit12.zip -d .

svn add \* --force

svn commit -m "r12 blue" --username=blue

echo "Revision 12 blue"

svn update

svn switch "$REPO\_URL/branches/branch2"

svn merge "$REPO\_URL/branches/branch3"

##

## ИСПРАВЛЕНИЕ КОНФЛИКТА ВРУЧНУЮ

##

svn add \* --force

echo "Слияние r6 и r12"

# r13 blue

unzip -o ../commits/commit13.zip -d .

svn add \* --force

svn commit -m "r13 blue" --username=blue

echo "Revision 13 blue"

svn update

svn switch "$REPO\_URL/trunk"

svn merge "$REPO\_URL/branches/branch2"

##

## ИСПРАВЛЕНИЕ КОНФЛИКТА ВРУЧНУЮ

##

svn add \* --force

echo "Слияние r11 и r13"

# r14 red

unzip -o ../commits/commit14.zip -d .

svn add \* --force

svn commit -m "r14 red" --username=red

echo "Revision 14 red"

svn update

# Вывод

В ходе выполнения лабораторной работы я улучшил свое владение системой контроля версий Git. Во время выполнения работы я научился настраивать репозитории Git и SVN, работать с ревизиями в соответствии с блок-схемой. Были изучены некоторые способы разрешения конфликтов.