

FACULTATEA CALCULATOARE, INFORMATICA SI MICROELECTRONICA

UNIVERSITATEA TEHNICA A MOLDOVEI

MEDII INTERACTIVE DE DEZVOLTARE A PRODUSELOR SOFT

LUCRAREA DE LABORATOR#2

Version Control Systems si modul de setare a unui server

Autor:

Iana PUSCAS TI-145

lector asistent:

Irina COJANU

lector superior:

Svetlana COJOCARU

Laboratory work #2

1 Scopul lucrarii de laborator

Studiarea Version Control Systems si modului de setare a unui server.

2 Obiective

- Intelegerea si folosirea CLI (basic level)
- Administrarea remote a masinilor linux machine folosind SSH (remote code editing)
- Version Control Systems (git || mercurial || svn)
- Compileaza codul C/C++/Java/Python prin intermediul CLI, folosind compilatoarele gcc/g++/javac/python

3 Laboratory work implementation

3.1 Tasks and Points

Advanced Level (nota 9 || 10):

- seteaza un branch to track a remote origin pe care vei putea sa faci push (ex. Github, Bitbucket or custom server)

- reseteaza un branch la commit-ul anterior
- merge 2 branches
- rezolvarea conflictelor a 2 branches

Bonus Point:

- Scribe un script care va compila HelloWorldPrograms projects.

3.2 Analiza lucrarii de laborator

Ссылка на репозиторий <https://github.com/IanaPushcash/MIDPS>.

В первой части лабораторной работы нами был создан репозиторий на сервере github.com. Далее с ним была связана папка на нашем компьютере и идентифицирован ключ SSH.

```
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS (master)
$ ls -l
итого 7
-rw-r--r-- 1 Яна 197121 0 фев 18 08:59 '# MIDPS'
-rw-r--r-- 1 Яна 197121 131 мар 20 08:45 aaaaaa.txt.txt
-rw-r--r-- 1 Яна 197121 7 фев 18 09:16 bbbb
drwxr-xr-x 1 Яна 197121 0 мар 29 16:55 forScript/
drwxr-xr-x 1 Яна 197121 0 мар 21 02:54 Lab#1/
drwxr-xr-x 1 Яна 197121 0 мар 21 02:53 Lab#3/
-rw-r--r-- 1 Яна 197121 1 фев 18 08:59 README.md

Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS (master)
$ git status
На ветке master
Ваша ветка обновлена в соответствии с «origin/master».
Изменения, которые не в индексе для коммита:
(используйте «git add/rm <файл>...», чтобы добавить или удалить файл из индекса)
(используйте «git checkout -- <файл>...», чтобы отменить изменения
в рабочем каталоге)

удалено:      HelloWorld.class
удалено:      hello.exe
удалено:      hello.pyc
удалено:      helloC++.cpp
удалено:      helloC++.exe
удалено:      helloC.c
удалено:      helloC.exe
удалено:      helloJ.java
удалено:      helloPY.py
удалено:      helloR8.rb
удалено:      script.sh

Неотслеживаемые файлы:
(используйте «git add <файл>...», чтобы добавить в то, что будет включено в комм
ит)

forScript/

нет изменений добавленных для коммита
(используйте «git add» и/или «git commit -a»)
```

Рис. 3.1– Проверим состояние репозитория.

```
SDK-64:/D/1 ???????/2 ???????/MIDPS
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS (master)
$ git add -A
warning: LF will be replaced by CRLF in forScript/script.sh.
The file will have its original line endings in your working directory.

Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS (master)
$ git commit -m "for script"
[master b9dd1d1] for script
warning: LF will be replaced by CRLF in forScript/script.sh.
The file will have its original line endings in your working directory.
11 files changed, 0 insertions(+), 0 deletions(-)
rename HelloWorld.class => forScript/HelloWorld.class (100%)
rename hello.pyc => forScript/hello.pyc (100%)
rename helloC++.cpp => forScript/helloC++.cpp (100%)
rename helloC++.exe => forScript/helloC++.exe (99%)
rename helloC.c => forScript/helloC.c (100%)
rename helloC.exe => forScript/helloC.exe (99%)
rename helloJ.java => forScript/helloJ.java (100%)
rename helloPY.py => forScript/helloPY.py (100%)
rename helloRB.rb => forScript/helloRB.rb (100%)
rename script.sh => forScript/script.sh (100%)
delete mode 100644 hello.exe

Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS (master)
$ git push
warning: push.default не установлен; его неявное значение было изменено в Git ве
рсии 2.0 с «matching» на «simple». Чтобы прекратить вывод этого сообщения и сохр
анить старое поведение, используйте:

    git config --global push.default matching

Чтобы прекратить вывод этого сообщения и использовать новое поведение, используйте:

    git config --global push.default simple

Когда push.default установлено в «matching», git будет отправлять изменения лока
льных веток в существующие внешние ветки с таким же именем.

Начиная с Git версии 2.0, по умолчанию используется более консервативное поведен
ие «simple», которое отправляет изменения текущей ветки в соответствующую внешнюю
ветку, из которой «git pull» забирает изменения.

Смотрите «git help config» и ищите «push.default» для дополнительной информации.
(режим «simple» появился в Git версии 1.7.11. Используйте похожий режим «current»
вместо «simple», если вы иногда используете старые версии Git)

Counting objects: 5, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 50.53 KiB | 0 bytes/s, done.
Total 5 (delta 2), reused 0 (delta 0)
To git@github.com:IanaPushcash/MIDPS.git
 7892b50..b9dd1d1 master -> master
```

Рис. 3.2– Добавим изменения на индексирование. Закомментируем изменения. Загрузим на репозиторий.

```
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS (master)
$ git status
На ветке master
Ваша ветка обновлена в соответствии с «origin/master».
ничего коммитить, нет изменений в рабочем каталоге
```

Рис. 3.3– Проверим состояние репозитория.

```
SDK-64:/D/1 ??????/2 ??????/MIDPS
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS (master)
$ git branch mybranch
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS (master)
$ git checkout mybranch
Переключено на ветку «mybranch»
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS (mybranch)
$ |
```

Рис. 3.4 – Создадим новую ветвь и перейдем на неё.

```
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS (mybranch)
$ cd forScript/
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS/forScript (mybranch)
$ ls
hello.pyс helloC.exe* helloC++.exe* helloPY.py HelloWorld.class
helloC.c helloC++.cpp helloJ.java helloRB.rb script.sh
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS/forScript (mybranch)
$ vi helloC.c
[Изменения не сохранены]
/usr/bin/bash: q: команда не найдена
Оболочка завершила работу 127
Нажмите ENTER или введите команду для продолжения
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS/forScript (mybranch)
$ cat helloC.c
#include <stdio.h>

int main()
{
    printf("what do you want to speak?\n");
    printf("Hello mworld\n");
    return 0;
}
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS/forScript (mybranch)
$ |
```

Рис. 3.5 – Изменим один из файлов в репозитории, создав этим конфликт.

```
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS/forScript (mybranch)
$ git add -A
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS/forScript (mybranch)
$ git commit -m "changes"
[mybranch dddc5d3] changes
1 file changed, 2 insertions(+), 1 deletion(-)
```

Рис. 3.6 – Добавим изменения на репозиторий.

```
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS/forScript (mybranch)
$ git checkout master
Переключено на ветку «master»
Ваша ветка обновлена в соответствии с «origin/master».
Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS/forScript (master)
$ git merge mybranch
Автослияние forScript/helloC.c
КОНФЛИКТ (содержимое): Конфликт слияния в forScript/helloC.c
Не удалось провести автоматическое слияние; исправьте конфликты и сделайте коммит результата.
```

Рис. 3.7 – Вернемся на основную ветвь и соединим обе ветви.


```

Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS/forScript (master|MERGING)
$ cat helloC.c
#include <stdio.h>

int main()
{
<<<<<<< HEAD
    printf("Hello world\n");
=====
    printf("What do you want to speak\n");
    printf("Hello my world\n");
>>>>>>> mybranch
    return 0;
}

Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS/forScript (master|MERGING)
$ vi helloC.c

Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS/forScript (master|MERGING)
$ cat helloC.c
#include <stdio.h>

int main()
{
    printf("What do you want to speak\n");
    printf("Hello my world\n");
    return 0;
}

```

Рис. 3.8 – Просматриваем и исправляем конфликт.

```

Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS/forScript (master|MERGING)
$ git add .

Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS/forScript (master|MERGING)
$ git commit -m "merge mybranch"
[master b84c74b] merge mybranch

Яна@MINGW64 /D/1 семестр/2 семестр/MIDPS/forScript (master)
$ git push
warning: push.default не установлен; его неявное значение было изменено в Git ве
рсии 2.0 с «matching» на «simple». Чтобы прекратить вывод этого сообщения и сохр
анить старое поведение, используйте:

    git config --global push.default matching

Чтобы прекратить вывод этого сообщения и использовать новое поведение, используй
те:

    git config --global push.default simple

Когда push.default установлено в «matching», git будет отправлять изменения лока
льных веток в существующие внешние ветки с таким же именем.

Начиная с Git версии 2.0, по умолчанию используется более консервативное поведен
ие «simple», которое отправляет изменения текущей ветки в соответствующую внешнюю
ветку, из которой «git pull» забирает изменения.

Смотрите «git help config» и ищите «push.default» для дополнительной информации.
(режим «simple» появился в Git версии 1.7.11. Используйте похожий режим «current
» вместо «simple», если вы иногда используете старые версии Git)

Counting objects: 6, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (6/6), done.
Writing objects: 100% (6/6), 679 bytes | 0 bytes/s, done.
Total 6 (delta 3), reused 0 (delta 0)
To git@github.com:IanaPushcash/MIDPS.git
 1b8cd7a..b84c74b master -> master

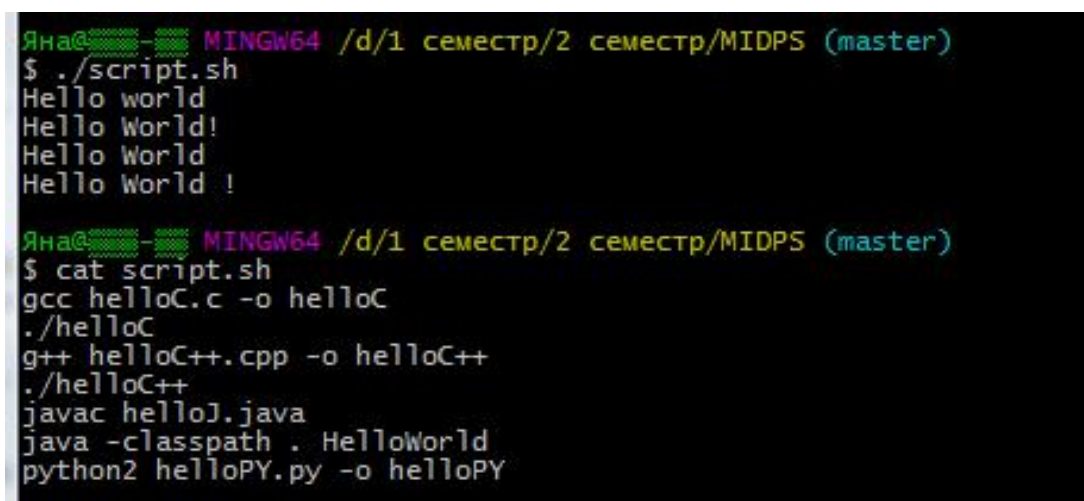
```

Рис. 3.9 – Добавляем изменения на репозиторий.



```
SDK-64:/d/1 ???????/2 ???????/MIDPS
Яна@MINGW64 /d/1 семестр/2 семестр/MIDPS (master)
$ gcc helloC.c -o helloC
Яна@MINGW64 /d/1 семестр/2 семестр/MIDPS (master)
$ ./helloC
Hello world
Яна@MINGW64 /d/1 семестр/2 семестр/MIDPS (master)
$ g++ helloC++.cpp -o helloC++
Яна@MINGW64 /d/1 семестр/2 семестр/MIDPS (master)
$ g++ helloC++.cpp -o helloC++
Яна@MINGW64 /d/1 семестр/2 семестр/MIDPS (master)
$ ./helloC++
Hello World!
Яна@MINGW64 /d/1 семестр/2 семестр/MIDPS (master)
$ javac helloJ.java
Яна@MINGW64 /d/1 семестр/2 семестр/MIDPS (master)
$ java -classpath . HelloWorld
Hello World
Яна@MINGW64 /d/1 семестр/2 семестр/MIDPS (master)
$ python2 helloPY.py -o helloPY
Hello World !
```

Рис. 3.10– Напишем скрипт для компиляции программ через Git Bash. Для этого подключим необходимые библиотеки и PATH. Для java прописываем export PATH=\$PATH:"/C/Program Files/Java/jdk1.8.0_60/bin/".



```
Яна@MINGW64 /d/1 семестр/2 семестр/MIDPS (master)
$ ./script.sh
Hello world
Hello World!
Hello World
Hello World !
Яна@MINGW64 /d/1 семестр/2 семестр/MIDPS (master)
$ cat script.sh
gcc helloC.c -o helloC
./helloC
g++ helloC++.cpp -o helloC++
./helloC++
javac helloJ.java
java -classpath . HelloWorld
python2 helloPY.py -o helloPY
```

Рис. 3.11– Объединим все команды в один скрипт.

Concluzie

В ходе данной лабораторной работы мною были изучены основы работы систем версионного контроля. Также я приобрела навыки работы с сервером из командной строки. Все полученные навыки являются актуальными и необходимыми в ежедневной работе программиста. Любой большой проект разрабатывается при использовании Git или SVN. Это помогает взаимодействию между разработчиками, а также дает возможность восстанавливать предыдущие версии файлов. Для полноценной работы с Git или SVN необходимы знания командной строки. Также можно использовать GUI версию, что намного легче и понятнее, но при этом возможна некоторая ограниченность функционала. Работа через командную строку является признаком профессионализма, а так же обеспечивает программисту больше возможностей работы с сервером.

Список литературы

- 1 Google, www.google.com
- 2 Stackoverflow, <http://ru.stackoverflow.com/>
- 3 Habrahabr, "Про Git на пальцах", <https://habrahabr.ru/post/68341/>