

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:
Druta ALEXANDRU

lector asistent:
Irina COJANU

1 Lucrarea de laborator 2

1.1 Scopul lucrarii

Insusirea notiunii de Version Control Systems si a modului de setare a unui server.

1.2 Obiective

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.

2 Realizarea lucrarii de laborator:

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

adaugam si selectam un nou branch pe nume "qwertMD"

```
NEXUS@NEXUS-PC MINGW64 /d/MIDPS (master)
$ git branch
* master

NEXUS@NEXUS-PC MINGW64 /d/MIDPS (master)
$ git branch qwertMD

NEXUS@NEXUS-PC MINGW64 /d/MIDPS (master)
$ git branch
* master
  qwertMD

NEXUS@NEXUS-PC MINGW64 /d/MIDPS (master)
$ git checkout qwertMD
M .gitignore
Switched to branch 'qwertMD'

NEXUS@NEXUS-PC MINGW64 /d/MIDPS (qwertMD)
$ |
```

setam un branch to track remote origin si facem push

```
NEXUS@NEXUS-PC MINGW64 /d/MIDPS (qwertMD)
$ git init
Reinitialized existing Git repository in D:/MIDPS/.git/

NEXUS@NEXUS-PC MINGW64 /d/MIDPS (qwertMD)
$ git config --global user.name "DrutaAlexandru"

NEXUS@NEXUS-PC MINGW64 /d/MIDPS (qwertMD)
$ git config --global user.email "drutasandu@gmail.com"

NEXUS@NEXUS-PC MINGW64 /d/MIDPS (qwertMD)
$ git branch
  master
* qwertMD

NEXUS@NEXUS-PC MINGW64 /d/MIDPS (qwertMD)
$ git push --set-upstream origin qwertMD
Total 0 (delta 0), reused 0 (delta 0)
To https://github.com/DrutaAlexandru/MIDPS.git
 * [new branch]      qwertMD -> qwertMD
Branch qwertMD set up to track remote branch qwertMD from origin.
```

- reseteaza un branch la commit-ul anterior

verificam log-ul de comituri

```

NEXUS@NEXUS-PC MINGW64 /d/MIDPS (qwertMD)
$ git log --graph --all --decorate
* commit a66108e7e90ce9288786d3986289ce7ef3b914f9 (HEAD -> qwertMD, origin/qwertMD)
Author: DrutaAlexandru <drutasandu@gmail.com>
Date: Sat Mar 26 13:21:53 2016 +0200

    hello world

* commit 73fb0afdd6ca7788c8c02582ed6fcd84ae64d6e5
Author: DrutaAlexandru <drutasandu@gmail.com>
Date: Sat Mar 26 13:19:39 2016 +0200

    adaugare fisier pentru commit

```

selectam din lista de commituri la un commit anterior prin introducerea inceputului codului de commit

```

NEXUS@NEXUS-PC MINGW64 /d/MIDPS (qwertMD)
$ git checkout 73fb0afdd
Note: checking out '73fb0afdd'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -b with the checkout command again. Example:

    git checkout -b <new-branch-name>

HEAD is now at 73fb0af... adaugare fisier pentru commit

```

- merge 2 branches

facem merge la cele 2 branch-uri in unul singur master

```

NEXUS@NEXUS-PC MINGW64 /d/MIDPS ((a66108e...))
$ git checkout master
Previous HEAD position was a66108e... hello world
Switched to branch 'master'
Your branch is up-to-date with 'origin/master'.

NEXUS@NEXUS-PC MINGW64 /d/MIDPS (master)
$ git merge master qwertMD
Updating de041db..a66108e
Fast-forward
 .gitignore | 180 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
 Lab 2/Helloworld.py | 1 +
 2 files changed, 180 insertions(+), 1 deletion(-)
 create mode 100644 Lab 2/Helloworld.py

```

- rezolvarea conflictelor a 2 branches

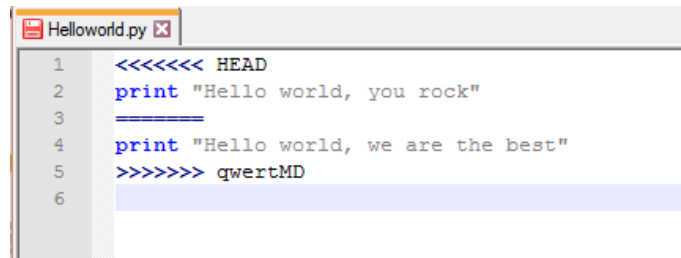
executam comanda de merge

```

NEXUS@NEXUS-PC MINGW64 /d/MIDPS (master|MERGING)
$ git merge qwertMD
error: merge is not possible because you have unmerged files.
hint: Fix them up in the work tree, and then use 'git add/rm <file>'
hint: as appropriate to mark resolution and make a commit.
fatal: Exiting because of an unresolved conflict.

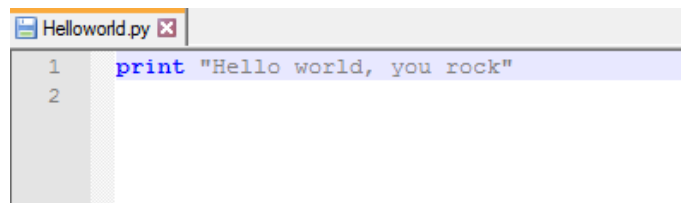
```

observam ca a aparut o eroare in fisierul Helloworld.py, astfel git a modificat fisierul sa alegem singuri care cod sa lasam



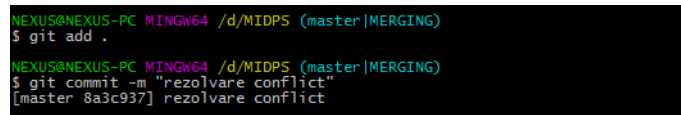
```
Helloworld.py
1 <<<<<< HEAD
2 print "Hello world, you rock"
3 =====
4 print "Hello world, we are the best"
5 >>>>>> qwertMD
6
```

am schimbat fisierul selectind codul care ne trebuie



```
Helloworld.py
1 print "Hello world, you rock"
2
```

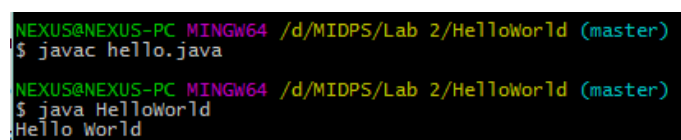
executam commit-ul de merge a celor 2 branch-urilor deja cu conflictul rezolvat



```
NEXUS@NEXUS-PC MINGW64 /d/MIDPS (master|MERGING)
$ git add .
NEXUS@NEXUS-PC MINGW64 /d/MIDPS (master|MERGING)
$ git commit -m "rezolvare conflict"
[master 8a3c937] rezolvare conflict
```

- Bonus point: Compilarea scripturilor HelloWorld projects

compilarea scriptului in java



```
NEXUS@NEXUS-PC MINGW64 /d/MIDPS/Lab 2/HelloWorld (master)
$ javac hello.java
NEXUS@NEXUS-PC MINGW64 /d/MIDPS/Lab 2/HelloWorld (master)
$ java HelloWorld
Hello World
```

compilarea scriptului in python

```
NEXUS@NEXUS-PC MINGW64 /d/MIDPS/Lab 2/HelloWorld (master)  
$ python hello.py  
Hello World !
```

3 Concluzie

In urma efectuării acestei lucrări de laborator am luat cunoștință cu mediul de lucru în echipă github. Astfel am efectuat task-uri de bază ca initializarea, setarea, accesul la github, adăugarea commiturilor, branchurilor, dar și diminuarea conflictelor, am executat scrierile de compilare și executare a programelor în java și python.

Aceste task-uri ne vor ajuta ulterior în programarea în echipă, astfel fiecare utilizator execută task-ul lui sau bucata lui de cod, după care încarcă lucrul efectuat pe server, iar la sfârșit tot proiectul se unește în unul singur și se efectuează produsul finit.