

# Лабораторна робота №2. Вступ до програмування. Основи de-bug процесу.

## 1 Вимоги

### 1.1 Розробник

- Муренко Лев Андрійович;
- студент групи КІТ-320;
- 28-ЖОВТ-2020.

### 1.2 Загальне завдання

Установити середовище для подальшої роботи з предмету “Програмування”.

## Хід роботи

1. Створюємо git репозиторій на сайті “github.com”.

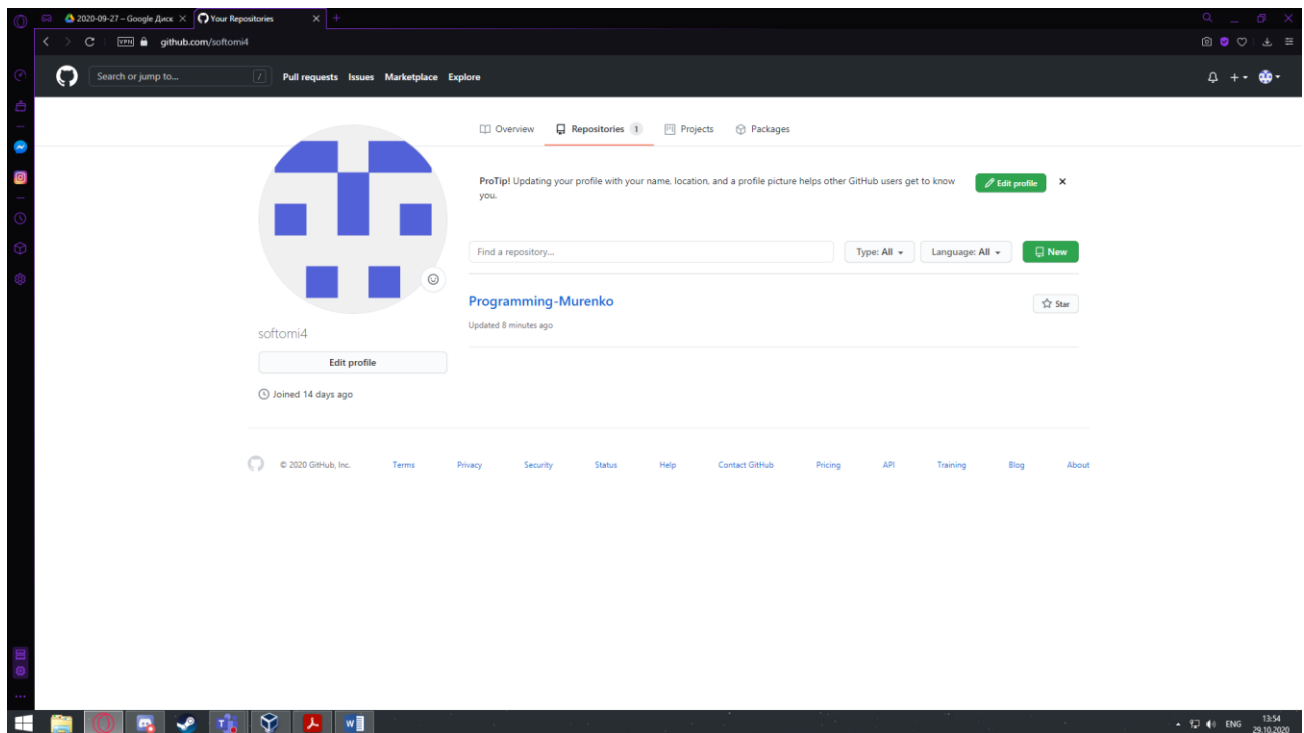
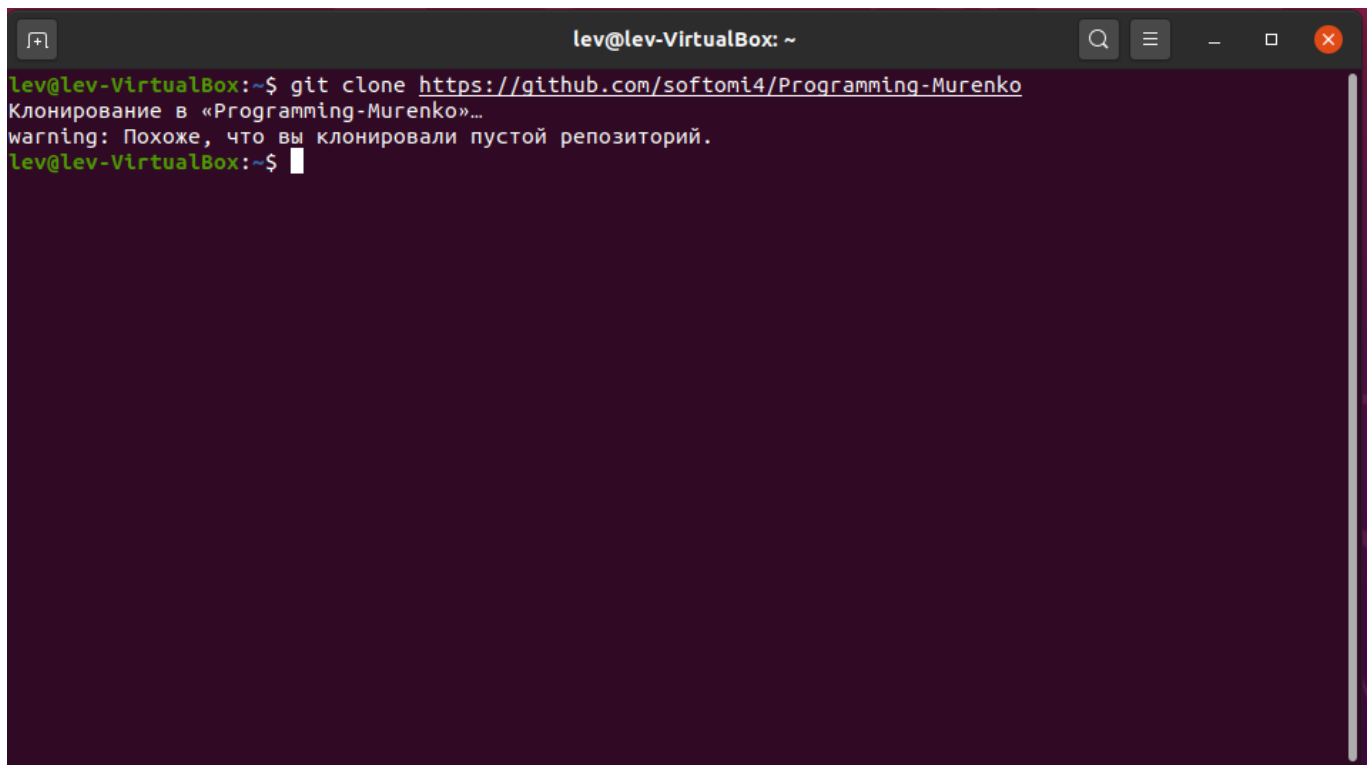


Рисунок 1 – Створений Репозиторій

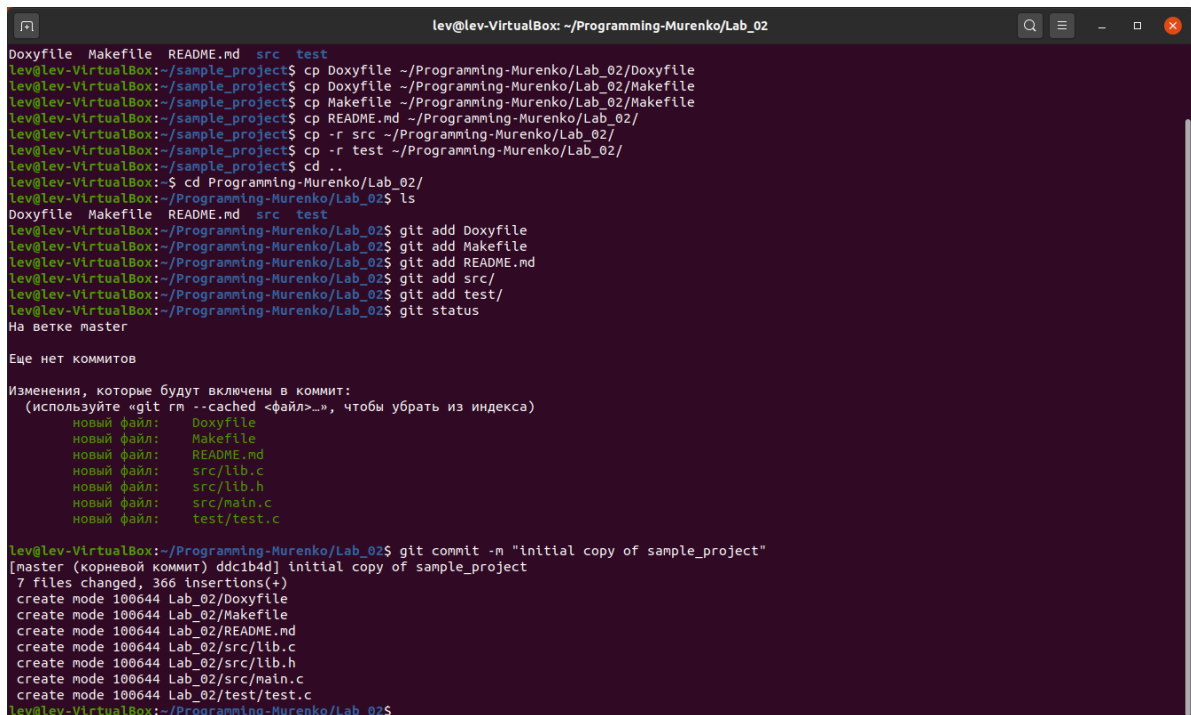
2. Склонуємо свій репозиторій з “github.com”.



```
lev@lev-VirtualBox: ~  
lev@lev-VirtualBox:~$ git clone https://github.com/softomi4/Programming-Murenko  
Клонирование в «Programming-Murenko»...  
warning: Похоже, что вы клонировали пустой репозиторий.  
lev@lev-VirtualBox:~$
```

Рисунок 2

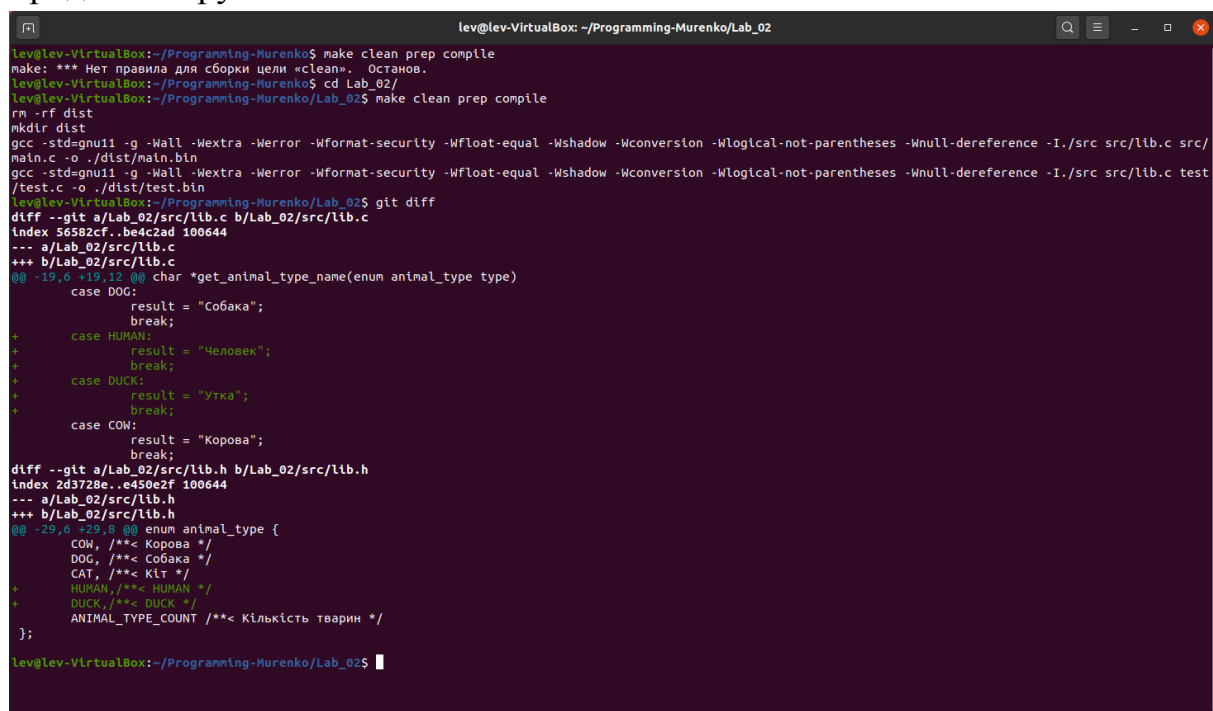
3. Склонуємо усі файли з зовнішнього репозиторія, що був наданий у попередній роботі у під-директорію Lab\_02. Зафіксуємо зміни під назвою “initial copy of sample\_project”.



```
lev@lev-VirtualBox: ~/Programming-Murenko/Lab_02  
Doxyfile Makefile README.md src test  
lev@lev-VirtualBox:~/sample_project$ cp Doxyfile ~/Programming-Murenko/Lab_02/Doxyfile  
lev@lev-VirtualBox:~/sample_project$ cp Doxyfile ~/Programming-Murenko/Lab_02/Makefile  
lev@lev-VirtualBox:~/sample_project$ cp Makefile ~/Programming-Murenko/Lab_02/Makefile  
lev@lev-VirtualBox:~/sample_project$ cp README.md ~/Programming-Murenko/Lab_02/  
lev@lev-VirtualBox:~/sample_project$ cp -r src ~/Programming-Murenko/Lab_02/  
lev@lev-VirtualBox:~/sample_project$ cp -r test ~/Programming-Murenko/Lab_02/  
lev@lev-VirtualBox:~/sample_project$ cd ..  
lev@lev-VirtualBox:~$ cd Programming-Murenko/Lab_02/  
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02$ ls  
Doxyfile Makefile README.md src test  
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02$ git add Doxyfile  
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02$ git add Makefile  
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02$ git add README.md  
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02$ git add src/  
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02$ git add test/  
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02$ git status  
На ветке master  
Еще нет коммитов  
Изменения, которые будут включены в коммит:  
(используйте «git rm --cached <файл>...», чтобы убрать из индекса)  
новый файл: Doxyfile  
новый файл: Makefile  
новый файл: README.md  
новый файл: src/lib.c  
новый файл: src/lib.h  
новый файл: src/main.c  
новый файл: test/test.c  
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02$ git commit -m "initial copy of sample_project"  
[master (корневой коммит) ddc1b4d] initial copy of sample_project  
7 files changed, 366 insertions(+)  
create mode 100644 Lab_02/Doxyfile  
create mode 100644 Lab_02/Makefile  
create mode 100644 Lab_02/README.md  
create mode 100644 Lab_02/src/lib.c  
create mode 100644 Lab_02/src/lib.h  
create mode 100644 Lab_02/src/main.c  
create mode 100644 Lab_02/test/test.c  
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02$
```

Рисунок 3

4. Виконуємо зміни на розсуд викладача, скомпілюємо проект та продемонструємо зміни.



```
lev@lev-VirtualBox: ~/Programming-Murenko/Lab_02
lev@lev-VirtualBox:~/Programming-Murenko$ make clean prep compile
make: *** Нет правил для сборки цели «clean». Останов.
lev@lev-VirtualBox:~/Programming-Murenko$ cd Lab_02/
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02$ make clean prep compile
rm -rf dist
mkdir dist
gcc -std=gnu11 -g -Wall -Wextra -Werror -Wformat-security -Wfloat-equal -Wshadow -Wconversion -Wlogical-not-parentheses -Wnull-dereference -I./src src/lib.c src/main.c -o ./dist/main.bin
gcc -std=gnu11 -g -Wall -Wextra -Werror -Wformat-security -Wfloat-equal -Wshadow -Wconversion -Wlogical-not-parentheses -Wnull-dereference -I./src src/lib.c test/test.c -o ./dist/test.bin
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02$ git diff
diff --git a/Lab_02/src/lib.c b/Lab_02/src/lib.c
index 56582cf..be4c2ad 100644
--- a/Lab_02/src/lib.c
+++ b/Lab_02/src/lib.c
@@ -19,0 +19,12 @@ char *get_animal_type_name(enum animal_type type)
     case DOG:
         result = "Собака";
         break;
+    case HUMAN:
+        result = "Человек";
+        break;
+    case DUCK:
+        result = "Утка";
+        break;
+    case COW:
+        result = "Корова";
+        break;
diff --git a/Lab_02/src/lib.h b/Lab_02/src/lib.h
index 2d3720c..c450e2f 100644
--- a/Lab_02/src/lib.h
+++ b/Lab_02/src/lib.h
@@ -29,0 +29,8 @@ enum animal_type {
     COW, /**< Корова */
     DOG, /**< Собака */
     CAT, /**< Кит */
+    HUMAN, /**< HUMAN */
+    DUCK, /**< DUCK */
     ANIMAL_TYPE_COUNT /**< Кількість тварин */
};
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02$
```

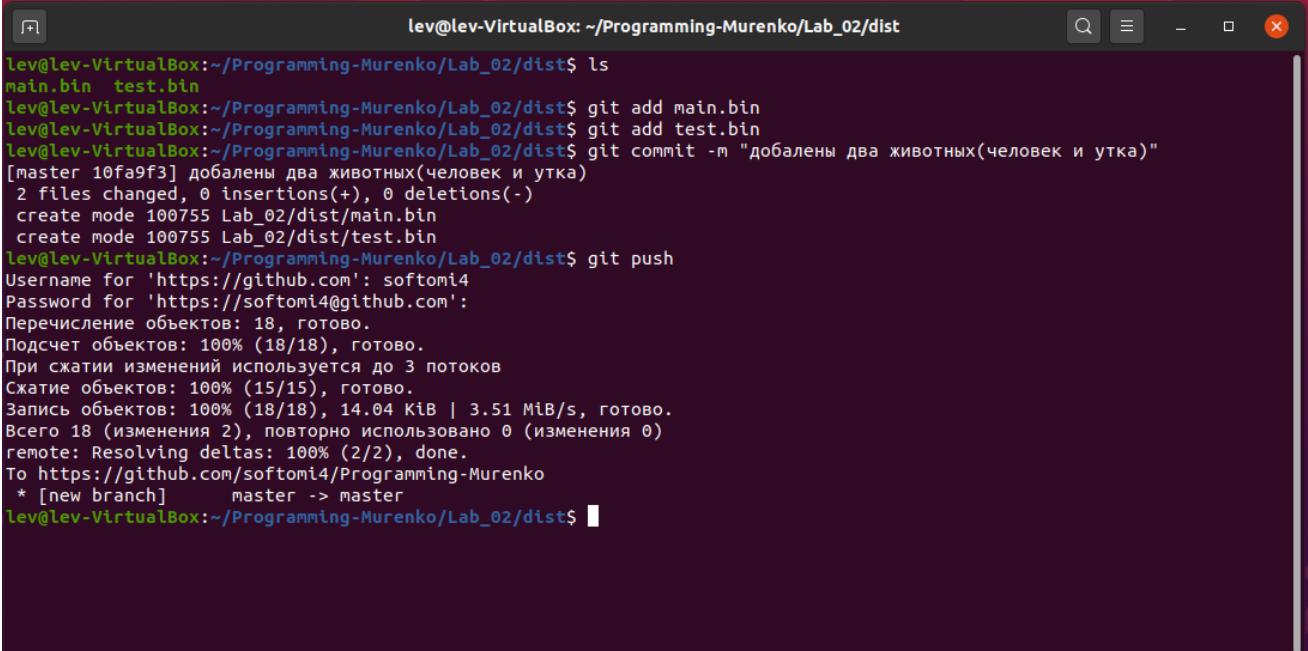
Рисунок 4

5. Step in – Выполнить следующую строку, заходя в функцию.  
Step out – Закончить выполнение текущей функции.  
Step over - Выполнить следующую строку, не входя в функцию.



Рисунок 5 – Функции Step over, Step in, Step out в Nemiver

6. Зафіксуємо зміни за допомогою команди `git commit` та завантажуюмо зміни за допомогою команди `git push`.



```
lev@lev-VirtualBox: ~/Programming-Murenko/Lab_02/dist
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02/dist$ ls
main.bin  test.bin
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02/dist$ git add main.bin
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02/dist$ git add test.bin
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02/dist$ git commit -m "добалены два животных(человек и утка)"
[master 10fa9f3] добалены два животных(человек и утка)
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100755 Lab_02/dist/main.bin
create mode 100755 Lab_02/dist/test.bin
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02/dist$ git push
Username for 'https://github.com': softomi4
Password for 'https://softomi4@github.com':
Перечисление объектов: 18, готово.
Подсчет объектов: 100% (18/18), готово.
При сжатии изменений используется до 3 потоков
Сжатие объектов: 100% (15/15), готово.
Запись объектов: 100% (18/18), 14.04 KiB | 3.51 MiB/s, готово.
Всего 18 (изменения 2), повторно использовано 0 (изменения 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/softomi4/Programming-Murenko
 * [new branch]      master -> master
lev@lev-VirtualBox:~/Programming-Murenko/Lab_02/dist$
```

Рисунок 6

7. Оптимізована компіляція перетворює початкову інформацію в дані які зрозумілі обчислюваній машині. А компіляція з інформацією про відлагодження поетапно демонструє процес проходження програми та допомагає виявити помилки.

## Висновки

При виконанні даної лабораторної роботи було набуто практичного досвіду роботи з репозиторієм та функціями Step in, Step out, Step over в програмі Nemiver.