МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ

Кафедра програмного забезпечення автоматизованих систем

ЗВІТ

про виконання лабораторної роботи № 3

з дисципліни

“ПРОЕКТНИЙ ПРАКТИКУМ”

­­­­­­­­­

|  |  |
| --- | --- |
| Перевірив:  *Ст. викладач*  *Журавель П.Д.* | Виконав:  Студент 2 курсу  Групи № ПЗ-144  *Даценко О.М.* |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

м. Черкаси

2015

**ЛАБОРАТОРНА РОБОТА №3**

**Тема:** сервіси спільної розробки проектів.

**Мета:** набути навиків роботи із сервісами хостингу IT-проектів на прикладі GitHub.

**ЗАВДАННЯ**

1. Перейти за адресою <https://github.com>, зареєструвати новий аккаунт і створити новий репозиторій.
2. Отримати URL новоствореного репозиторію і на локальному комп’ютері засобами Source Tree створити клон цього репозиторію.
3. У локальному клоні репозиторію створити новий проект, призначений для вирішення задачі відповідно до індивідуального завдання. Виконати комміт.
4. Розв’язати задачу відповідно до індивідуального завдання. Створити комміт.
5. Відправити зміни в локальному репозиторії на хост GitHub.
6. Зайти у віддалений репозиторій задачі за допомогою веб-інтерфейсу і внести зміни до проекту (додати коментарі до розв’язаного завдання).
7. Засобами Source Tree отримати зміни (*pull*) з віддаленого репозиторію.

У звіт з лабораторної роботи включити опис всіх дій, текст програми вирішення задачі, результати її роботи, а також скріншоти для ілюстрації основних операцій над репозиторіями.

**Індивідуальне завдання**

**Варіант 4**

Написати функцію, яка видаляє з дерева всі парні елементи.

**ВИКОНАННЯ ЗАВДАННЯ**

**Варіант 4**

Написати функцію, яка видаляє з дерева всі парні елементи.

**Лiстинг**

#include<iostream>

#include<iomanip>

#include<algorithm>

#include<stdlib.h>

#include<time.h>

#include<locale>

using namespace std;

struct Node {

int value;

Node \*left;

Node \*right;

};

void InsertTree(Node \*\*aNode, int data) {

if (!(\*aNode)) {

(\*aNode) = new Node;

(\*aNode)->value = data;

(\*aNode)->left = NULL;

(\*aNode)->right = NULL;

}

else {

if (data < (\*aNode)->value) InsertTree(&(\*aNode)->left, data);

else InsertTree(&(\*aNode)->right, data);

}

return;

}

void PrintTree(Node \*aNode) {

if (aNode) {

PrintTree(aNode->left);

cout << aNode->value << " ";

PrintTree(aNode->right);

}

return;

}

void Del(Node \*\*r, Node \*\*q)

{

Node \*s;

if (!(\*r)->right)

{

(\*q)->value = (\*r)->value;

\*q = \*r;

s = \*r; \*r = (\*r)->left; delete s;

}

else Del(&((\*r)->right), q);

}

void Delete(Node \*\*p, int data)

{

Node \*q;

if (!p) cout << "Дерево не знайдено!" << endl;

else

if (data<(\*p)->value) Delete(&((\*p)->left), data);

else

if (data>(\*p)->value) Delete(&((\*p)->right), data);

else

{

q = \*p;

if (!q->right)

{

\*p = q->left; delete q;

}

else

if (!q->left) { \*p = q->right; delete q; }

else Del(&(q->left), &q);

}

return;

}

int gMax(Node \*aNode, int maximum) {

if (aNode == NULL) {

return maximum;

}

return max(gMax(aNode->left, aNode->value),

gMax(aNode->right, aNode->value));

}

int gMin(Node \*aNode, int minimum) {

if (aNode == NULL) {

return minimum;

}

return min(gMin(aNode->left, aNode->value),

gMin(aNode->right, aNode->value));

}

int gSum(Node \*aNode, int max, int min) {

return max + min;

}

int main()

{

setlocale(LC\_ALL, "Russian");

Node \*root = new Node;

root = NULL;

InsertTree(&root, 24);

InsertTree(&root, 5);

InsertTree(&root, 7);

InsertTree(&root, 56);

InsertTree(&root, 49);

InsertTree(&root, 32);

cout << "Елементи бiнарного дерева" << endl;

PrintTree(root);

cout << endl;

cout << "Max елемент = " << gMax(root, -100) << endl;

cout << "Min елемент = " << gMin(root, 100) << endl;

cout << "Сума заданих елементiв(max i min) = " << gSum(root, gMax(root, -100), gMin(root, 100)) << endl;

Delete(&root, 24);

Delete(&root, 5);

Delete(&root, 7);

Delete(&root, 56);

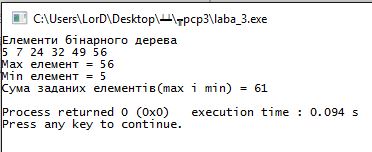
Delete(&root, 49);

Delete(&root, 32);

return 0;

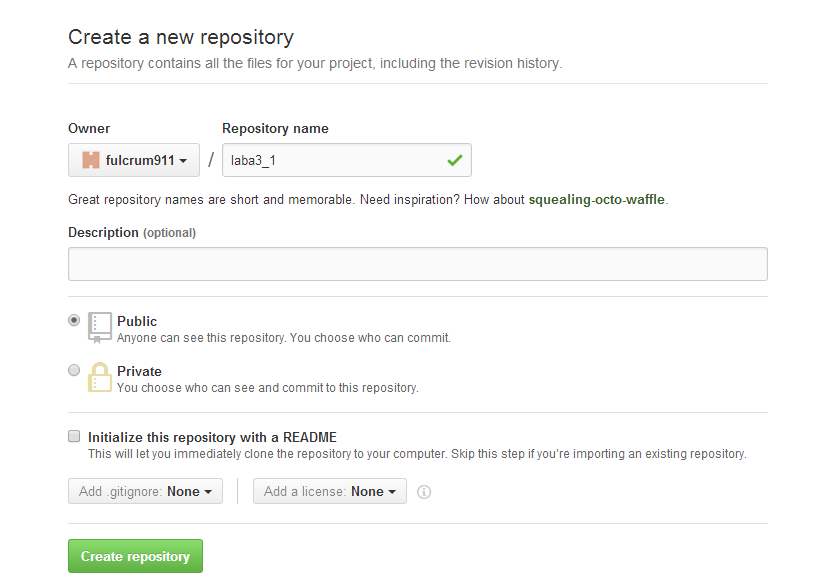
}

**Результат роботи програми № 1**

****

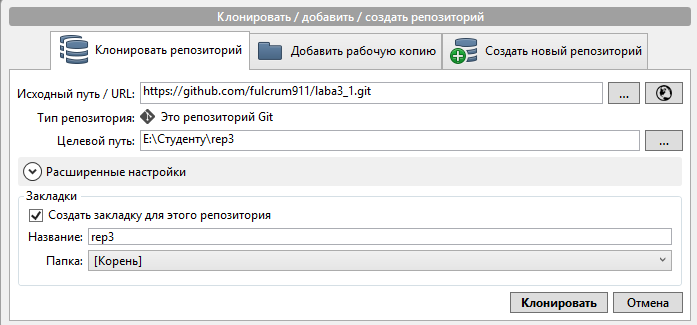
**Робота с Git**

1. Створив акаунт на https://github.com та віддалений репозиторію

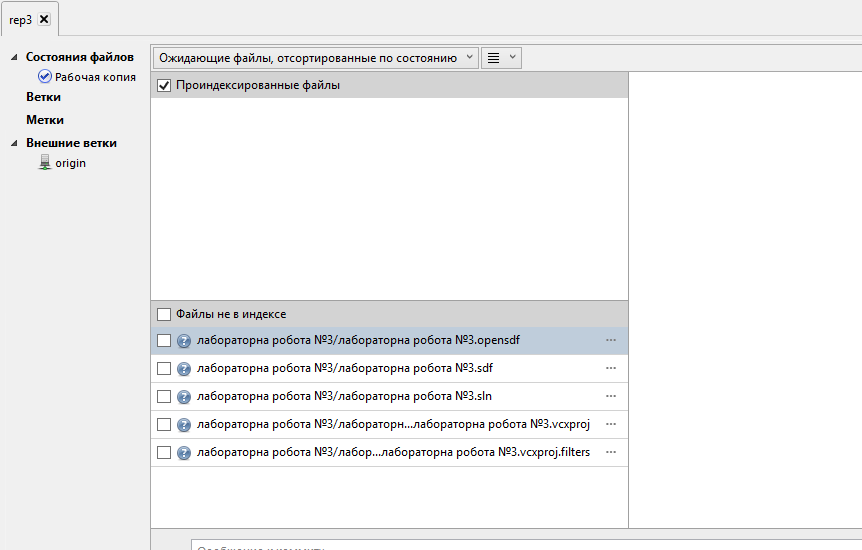


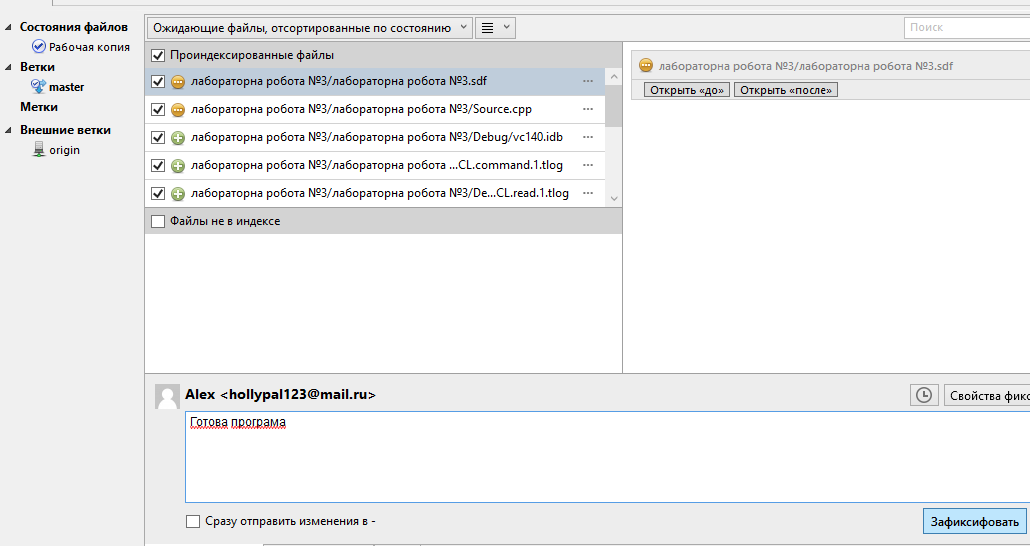
1. Отримав URL новоствореного репозиторію і на локальному комп’ютері

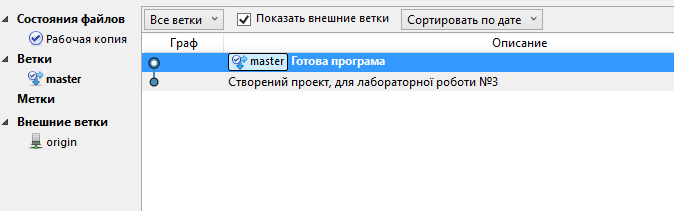
засобами Source Tree створив клон цього репозиторію.



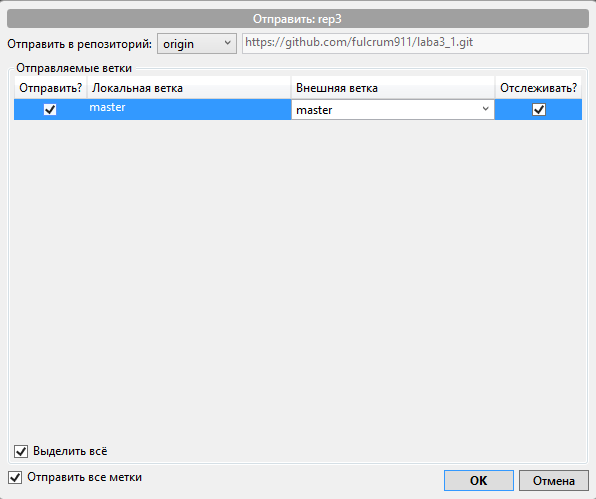
1. Виконуємо необхідні коміти



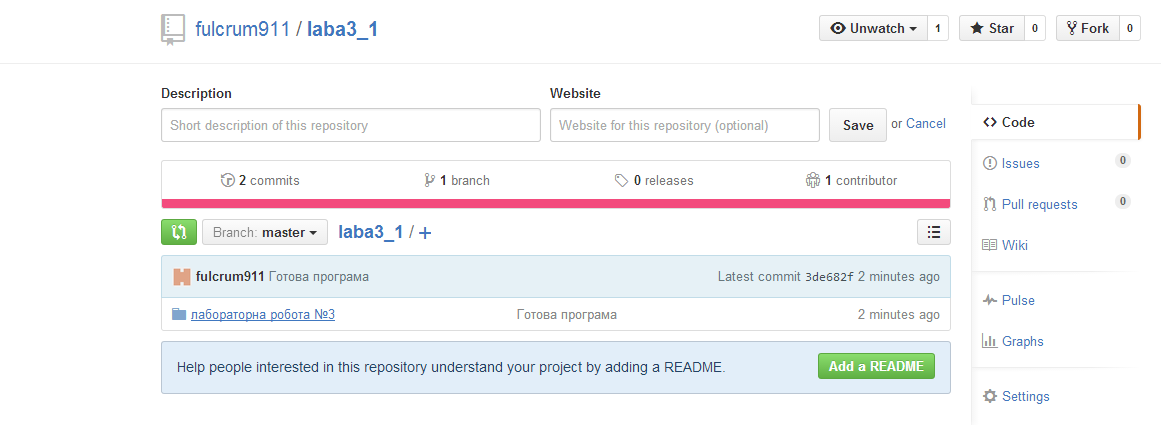




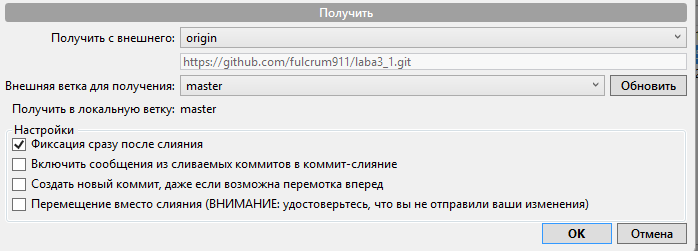
4. Відправив зміни в локальному репозиторії на хост GitHub.



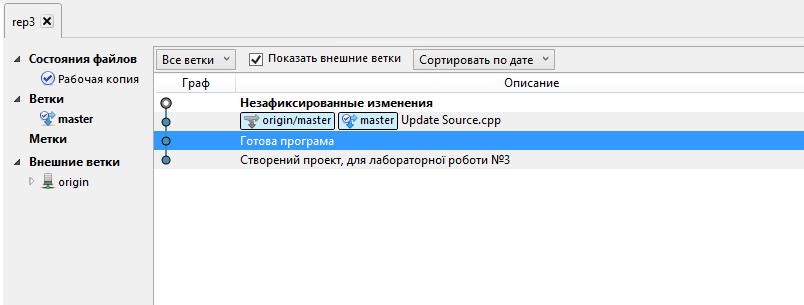
1. Зайшов у віддалений репозиторій задачі за допомогою веб-інтерфейсу і вніс зміни до проекту (додав коментарі до розв’язаного завдання).



1. Засобами Source Tree отримав зміни (pull) з віддаленого репозиторію.



1. Остаточний вигляд.



**Висновок:** Під час виконання даної лабораторної роботи я набув навичків роботи із сервісами хостингу IT-проектів на прикладі GitHub та ознайомився з бінарними деревами.