

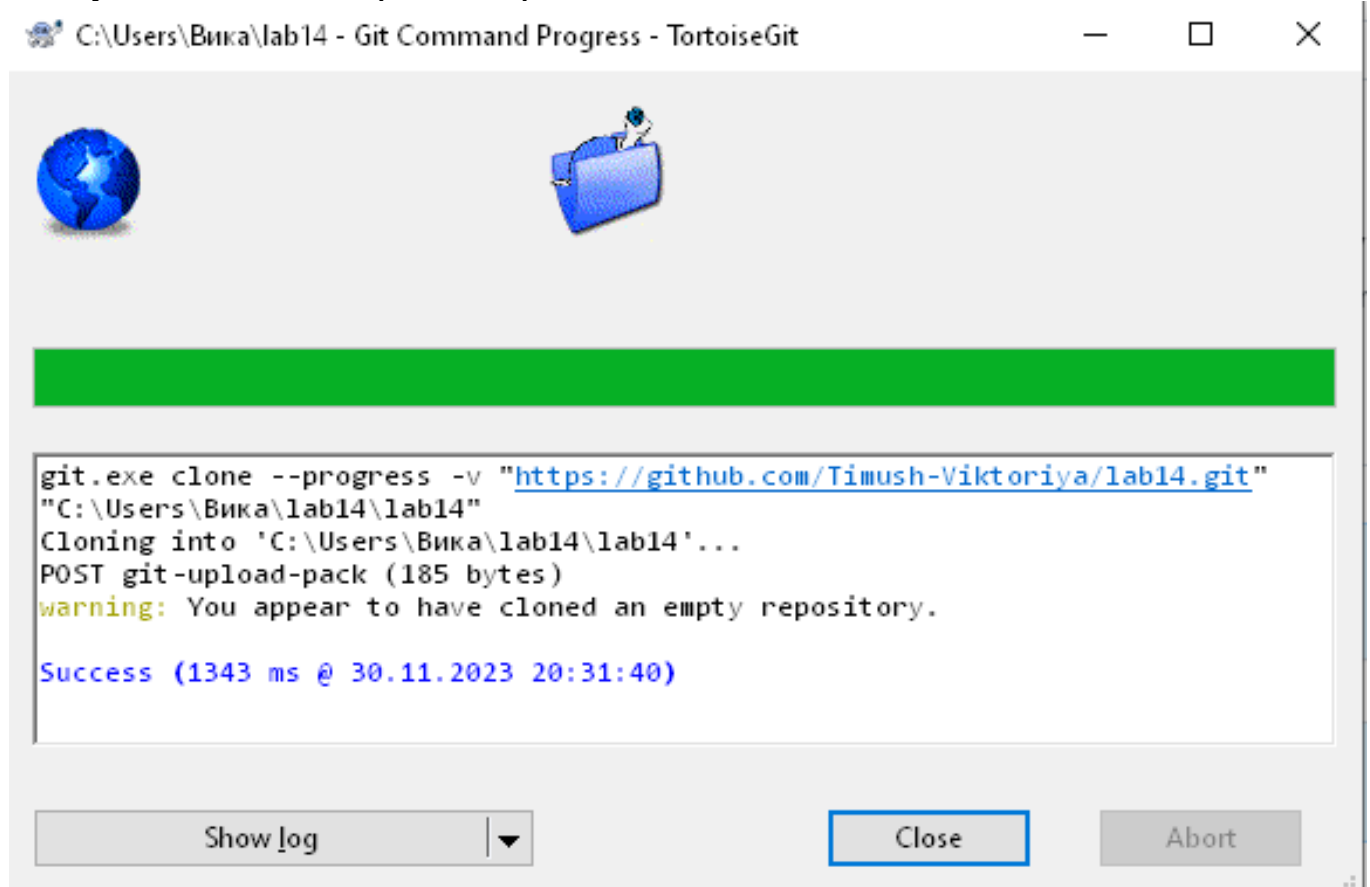
Лабораторна робота № 14
з дисципліни «Проектний практикум»

Тема: Робота з графічним клієнтом TortoiseGit

Мета роботи: Отримати навички роботи у графічній оболонці TortoiseGit

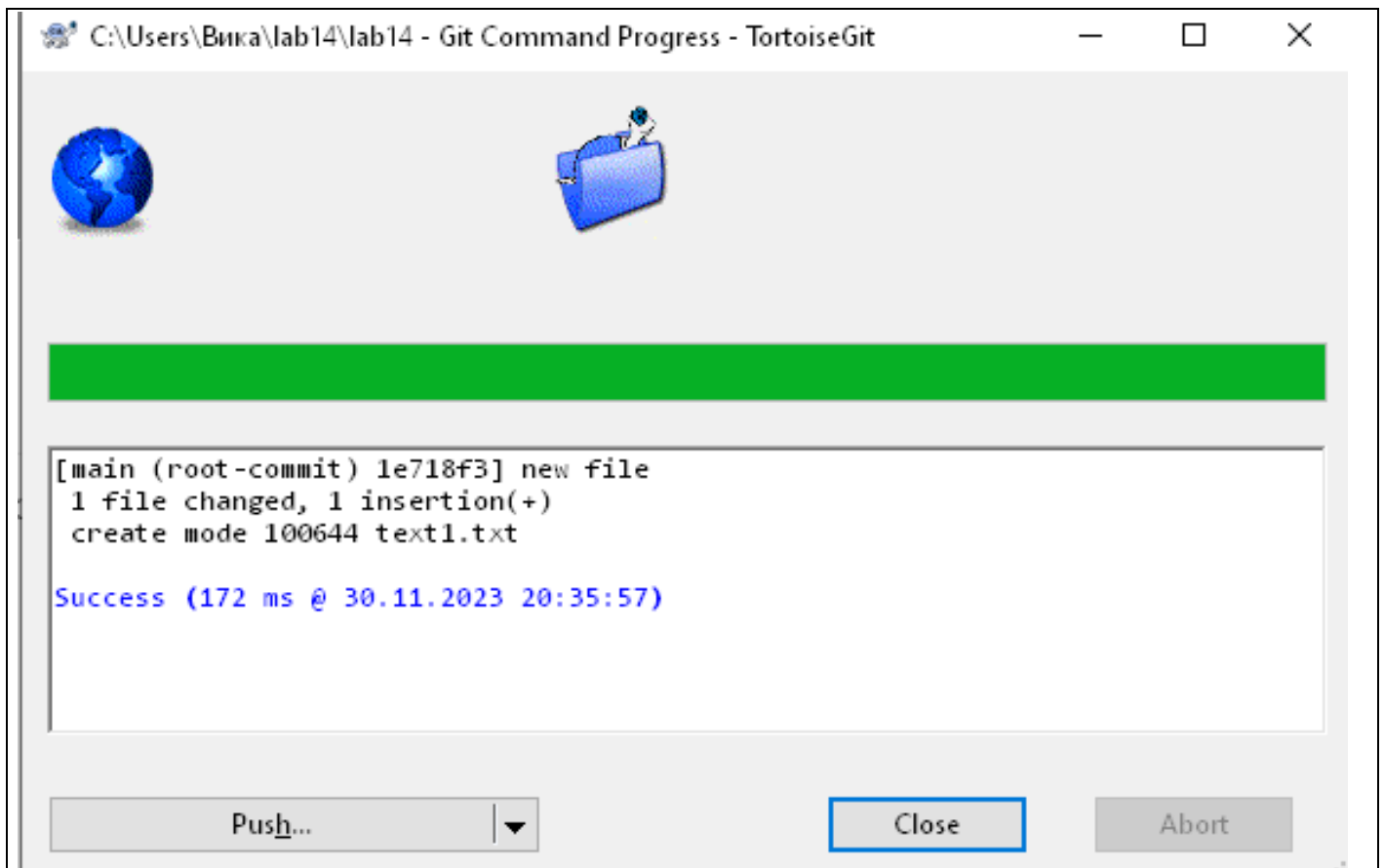
Хід роботи:

Клонувати віддалений репозиторій

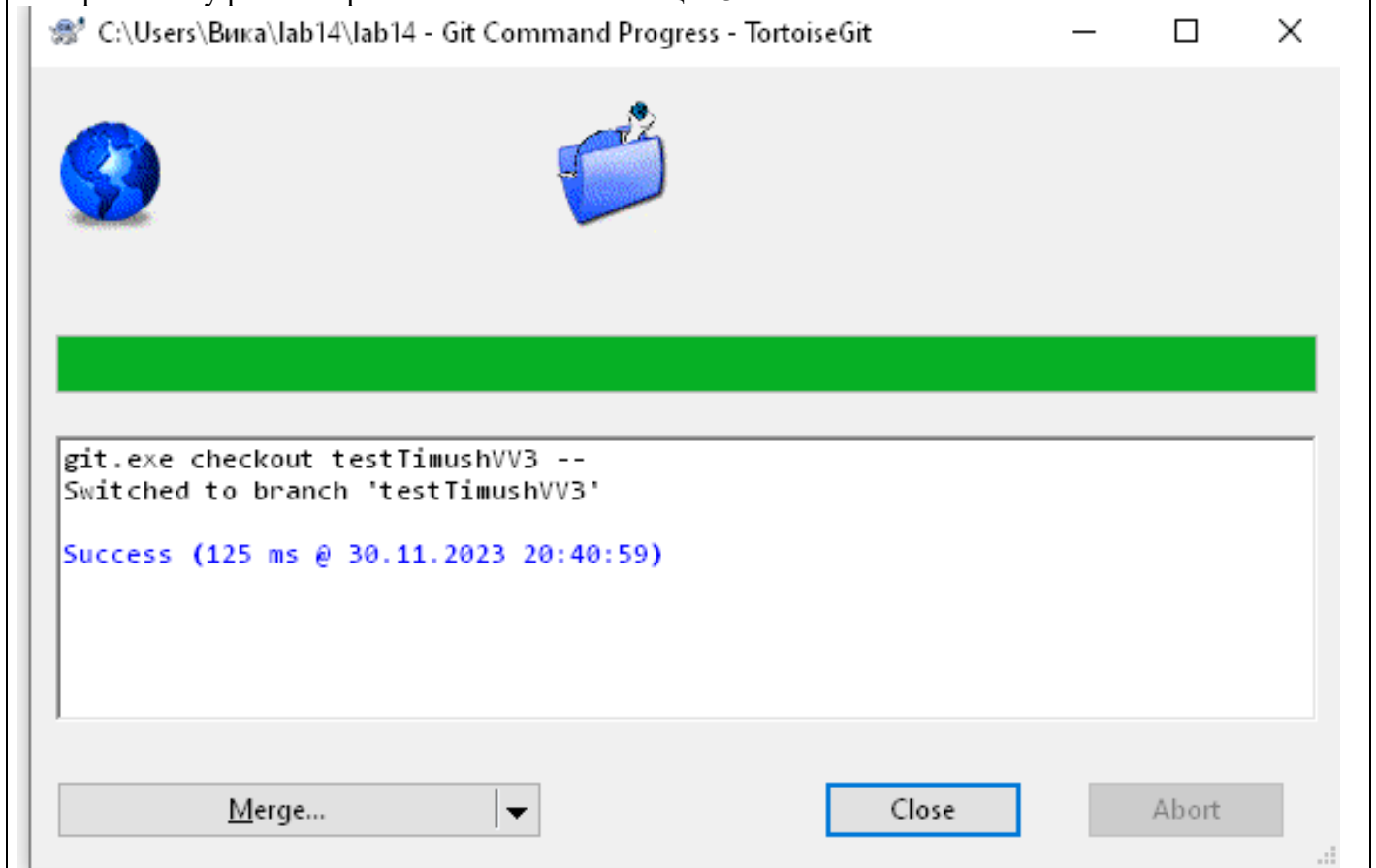


Внести зміни в репозиторій та зафіксувати їх.

					Лабораторна робота № 14										
Зм	Лист	№ докум	Підпис	Дата											
Розробив					Робота з графічним клієнтом TortoiseGit				Літ.		Лист.		Листів.		
Перевірив															
Оцінка															
Затв															
										Група451					

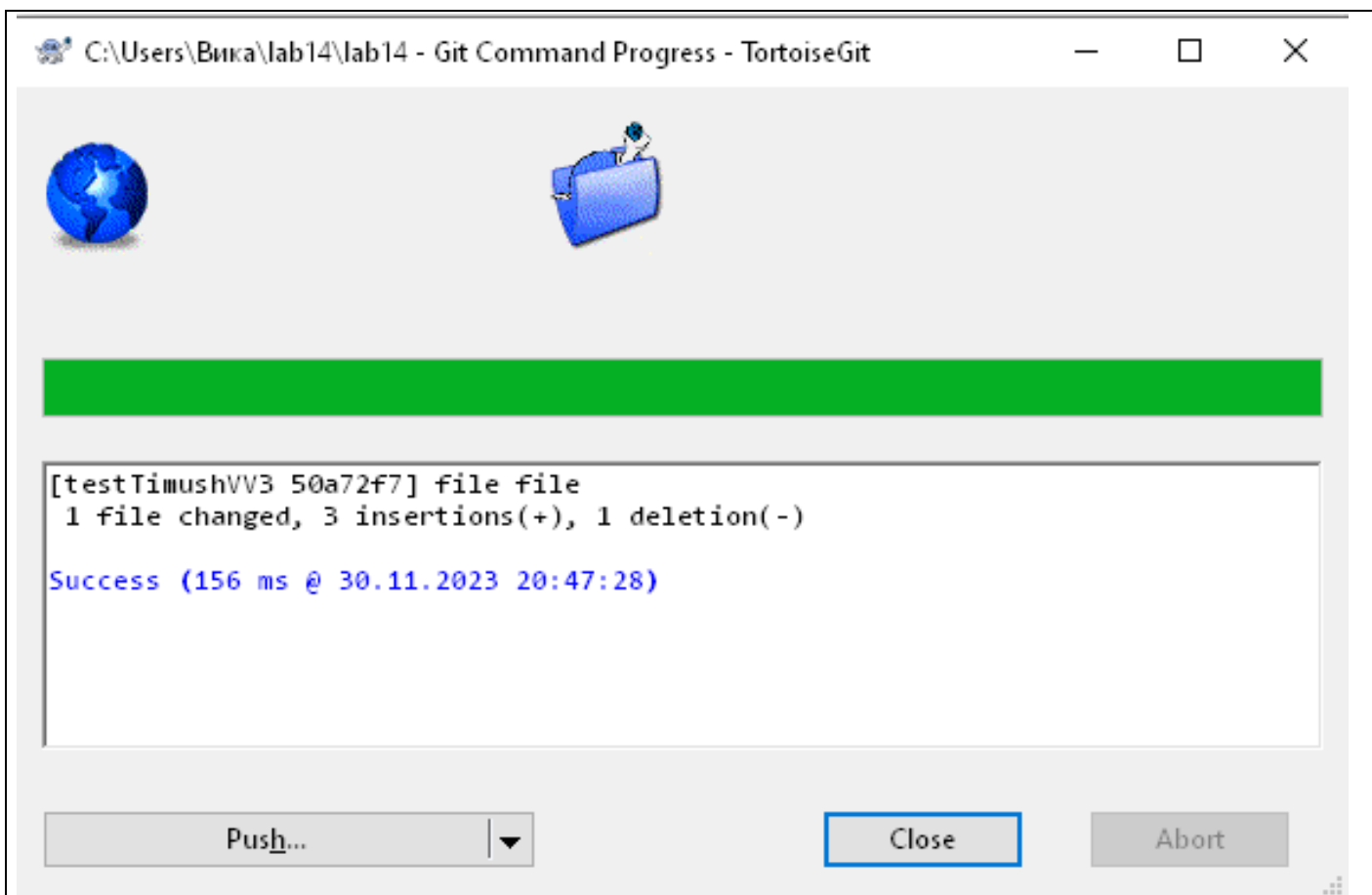


Створити гілку репозиторію "testВашіПБлатиницею3"

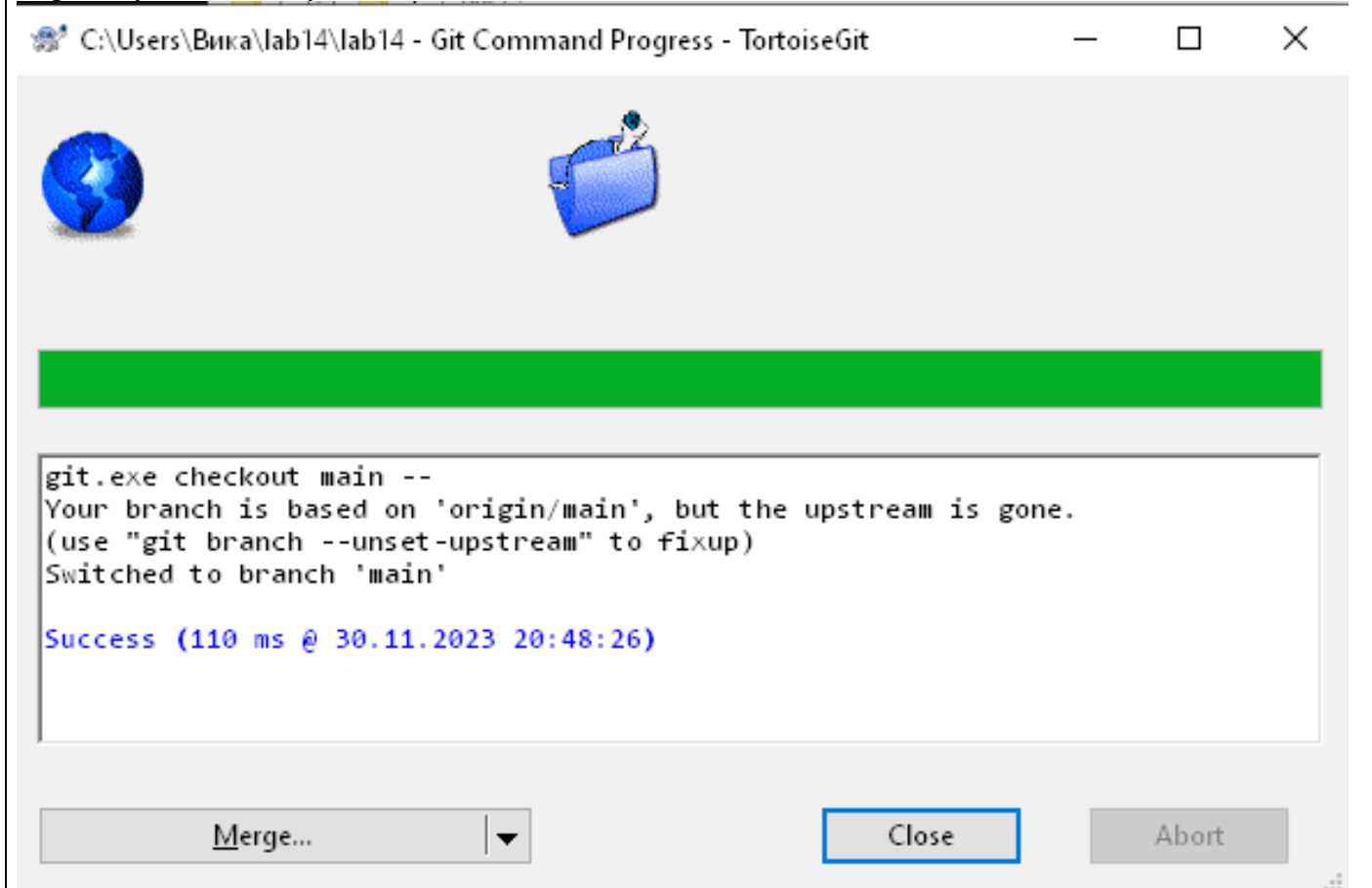


Внести зміни та зафаксувати їх

					Лабора т орна робота № 14	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		

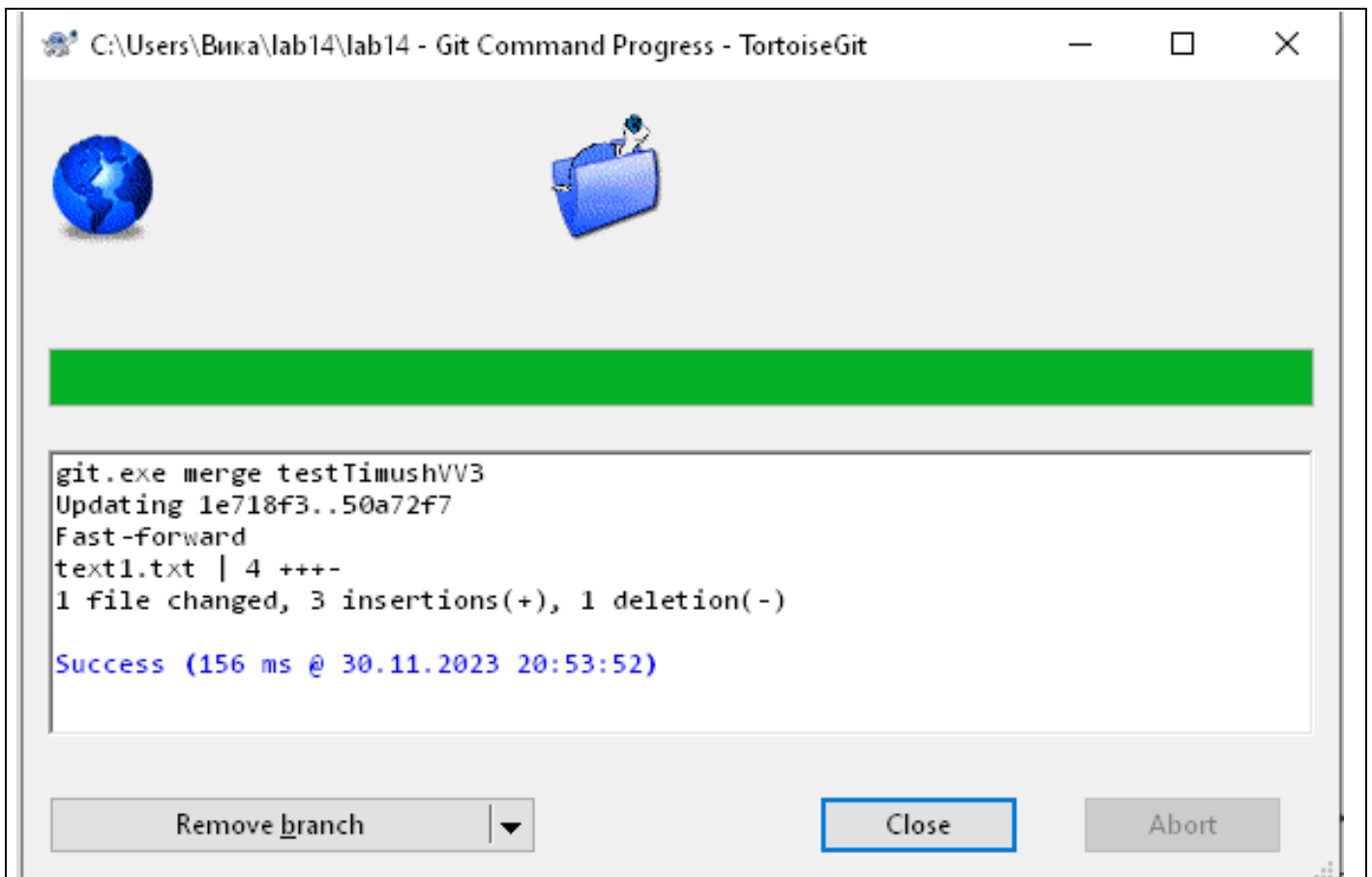


Перемкнутися на гілку "master"



Виконати злиття всіх існуючих гілок з гілкою "master"

					<p style="text-align: center;"><i>Лабораторна робота № 14</i></p>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		



Контрольні питання:

1.Що таке TortoiseGit?

TortoiseGit - це клієнтська програма для роботи з системою контролю версій Git. Вона надає графічний інтерфейс користувача (GUI) для взаємодії з Git, що полегшує використання цієї потужної системи управління версіями.

2.Для чого використовують TortoiseGit?

- 1.Клонування репозиторію
- 2.Фіксація змін (коміти)
- 3.Графічний перегляд історії комітів
- 4.Гілки і злиття
- 5.Робота з віддаленими репозиторіями
- 6.Керування ветками

3.Які вам відомі аналоги TortoiseGit?

TortoiseSVN

TortoiseHG

TortoiseCVS

SmartGit

GitKraken

					Лабораторна робота № 14	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		

4. Чи можливо використати графічний інтерфейс TortoiseGit для злиття гілок?

1. Відкрийте TortoiseGitMerge
2. Виберіть гілки
3. Виконайте злиття (Merge)
4. Фіксація злиття (Commit)
5. Підтвердження злиття (Push).

5. TortoiseGit: в чому різниця між "Git Sync", "Fetch" і "Pull"?

Git Sync:

"Git Sync" - це комбінація "Fetch" та "Pull". Ця опція виконує оновлення вашого локального репозиторію, отримуючи всі зміни з віддаленого репозиторію та об'єднуючи їх із вашим локальним кодом. Вона також автоматично вирішує можливі конфлікти.

Fetch:

"Fetch" завантажує всі зміни з віддаленого репозиторію, але не об'єднує їх із вашим локальним кодом. Це корисно, коли ви хочете переглянути зміни, які знаходяться в віддаленому репозиторії, до того, як об'єднувати їх із своїми змінами.

Pull:

"Pull" виконує "Fetch" та автоматично об'єднує зміни з віддаленого репозиторію із вашим локальним кодом. Це може призвести до конфліктів, якщо ви внесли зміни в тому ж самому місці, де були внесені інші зміни в віддаленому репозиторії.

					<i>Лабораторна робота № 14</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		