Міністерство освіти і науки України

Львівський національний університет імені Івана Франка

факультет Електроніки та комп’ютерних технологій

кафедра Радіофізики та комп’ютерних технологій

Звіт

про виконання лабораторної роботи №1-2

“ Структура файлової системи UNIX.

Система розмежування доступу в UNIX та Solaris ”

Виконав  
студент групи ФеІ-23

Дзюрах Іван Миколайович

Перевірив

ас. Сінькевич О.О

Львів – 2019

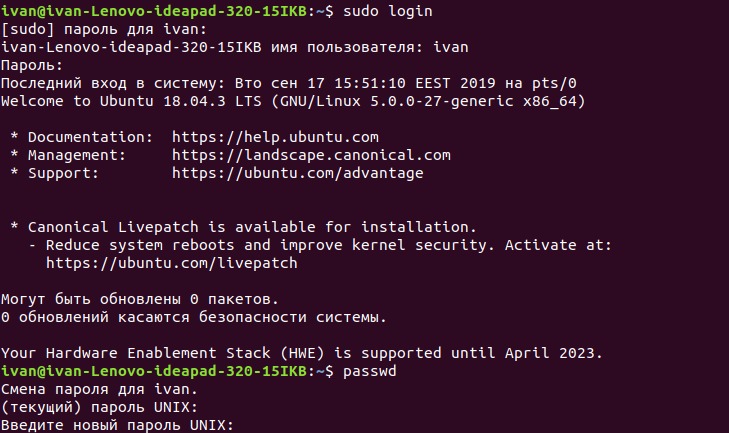
**Теоретичний матеріал**

Основна функція в довільній ОС – розділення простору зовнішньої пам’яті на іменовані ділянки – файли. В UNIX файли є послідовностями байтів. Іноді до них звертаються як до текстових або двійкових але розрізняються вони лише вмістом (текстові файли можуть містити тільки значення, що можливі в коді ASCII), а не структурою, або методом доступу. Файлова система UNIX ієрархічна, використовує багаторівневий метод зберігання файлів та створює ілюзію фізичного існування в рамках ОС. В UNIX – каталоги – це файли, які може мати довільна програма. Але в них зберігається не вся інформація про файли, а тільки імена файлів і числа, що використовуються ядром для доступу до тонких структур. Прихована частина файлової системи UNIX – це індексний дескриптор файлу. Існує по одному дескриптору на кожний файл і саме з ним працює файлова система. Індексний дескриптор описує розташування файлу; його довжину; метод доступу до нього; числа створення, модифікації; ідентифікатор власника та інше. Звичайний користувач не працює з індексними дескрипторами. Блок 0 – блок початкового завантаження, містить коротку програму–завантажувач. Звичайно вона читає більш довгу вторинну програму, завантажуючи тим самим ядро ОС UNIX. Блок 1– суперблок, містить заголовок файлової системи, що складається з основної інформації про неї, зокрема, про її розміри, число індексних дескрипторів, а також про параметри, що відносяться до списку вільних блоків. При монтуванні файлової системи, що виконується за командою mount, в таблиці змонтованих систем, розміщеній в ядрі, формується відповідний елемент, суперблок змонтованої системи записується в один з великих внутрішніх буферів ядра. Ядру повинні бути доступні суперблоки усіх змонтованих файлових систем, оскільки інформація, що в них знаходиться, дозволяє виконувати звертання до файлів та індексних дескрипторів. Індексні дескриптори зберігаються в файловій системі в усіх блока окрім 0. У залежності від розмірів, різні файлові системи містять різне число індексних дескрипторів (це число вказане в суперблоці). Оскільки дескриптори мають фіксований розмір та послідовно пронумеровані, починаючи з 0, довільний з них можна знайти за його номером. В UNIX розташування файла задається списком його блоків. Блоки можуть бути фізично розкидані на диску, але логічно вони створюють довгий ланцюжок, що містить весь набір даних. Ключ, що задає розташування файла, це список з 13 номерів блоків на диску, що зберігаються в індексному дескрипторі. Перші 10 номерів задаються перші 10 блоків файлу. Якщо дані файла розміщуються в цих 10 блоках або навіть не займають їх всі, то тільки декілька елементів списку вміщують дискові адреси. Наприклад: Якщо файлу достатньо чотирьох блоків, то перші чотири елементи списку включають відповідні адреси, а інші дев’ять заповнені нулями. В тому випадку коли довжина даних перевищує 10 блоків (5120 байт), перші десять елементів списку посилаються на блоки файла, а одинадцятий на блок, де розміщується список наступних 128 блоків файла. Цей блок називається посереднім. Для файлів довжина яких більша, ніж 138 (128 + 10) блоків (70656 байт), дванадцятий елемент списку вміщує адресу блоку, що включає адреси 128 посередніх блоків. Такий блок називається подвійним посереднім блоком. Розрахована на багато користувачів система UNIX підтримує механізм, відомий, як система прав доступу до файлів. Цей механізм дозволяє вказати для кожного файлу власника і забезпечити захист файлів від доступу інших користувачів. UNIX дозволяє також спільно використовувати файли декільком користувачам і групам користувачів. У більшості систем, за замовчуванням, іншим користувачам дозволяється читати ваші файли, але забороняється вносити зміни у файли або видаляти їх. Отже, кожен файл має конкретного власника, але, крім того файлами, також можуть володіти конкретні групи користувачів, які визначаються при реєстрації користувачів у системі. Кожен користувач стає членом як мінімум однієї групи користувачів (своєї або який-то із загальних груп). Системний адміністратор (суперкористувач, зазвичай root) може забезпечити користувачеві доступ більш, ніж до однієї групи.

**Виконання роботи**

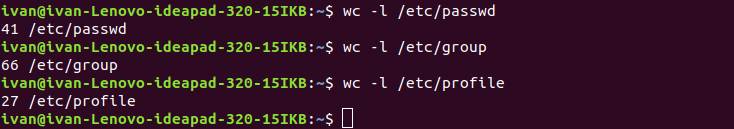
**Частина №1**

1. Завантажтеся в систему під вашим користувацьким ім'ям та поміняйте ваш пароль.

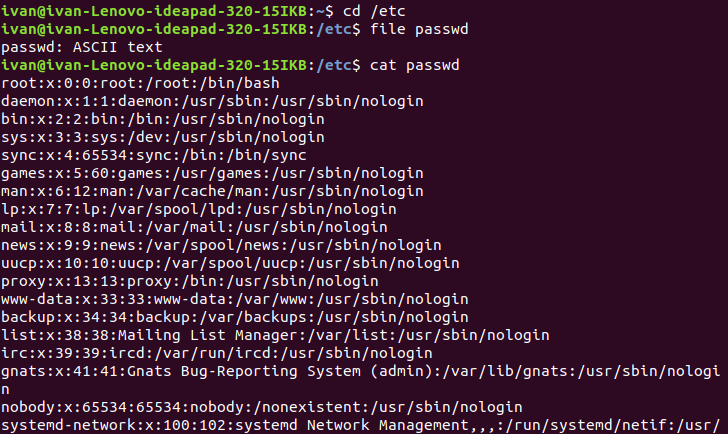
****

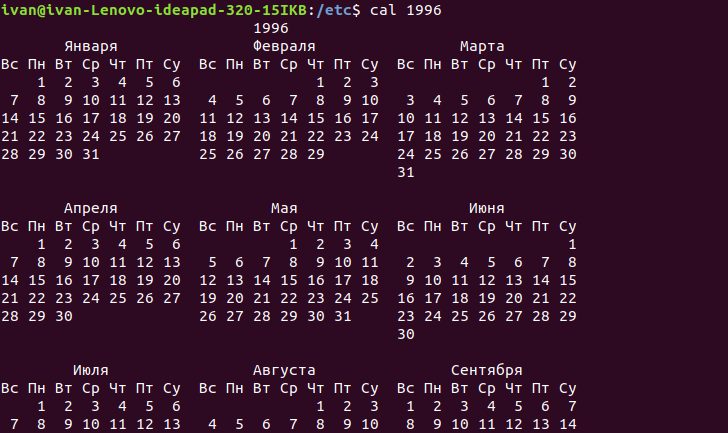
2. Виведіть системну дату. 

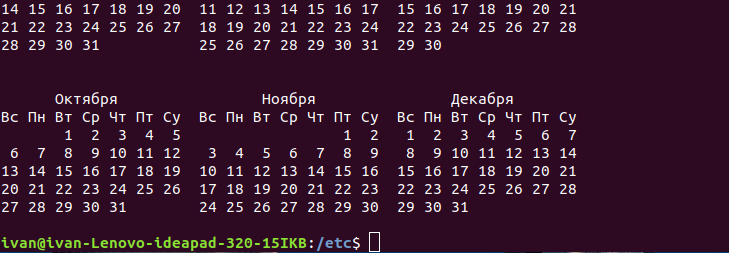
3. Підрахуйте кількість рядків у файлі.



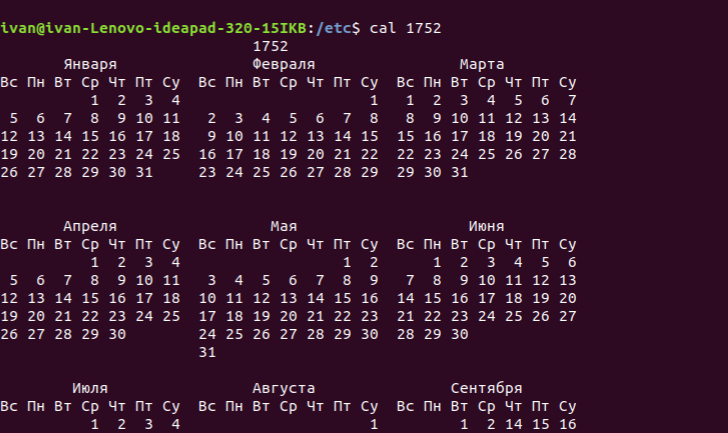
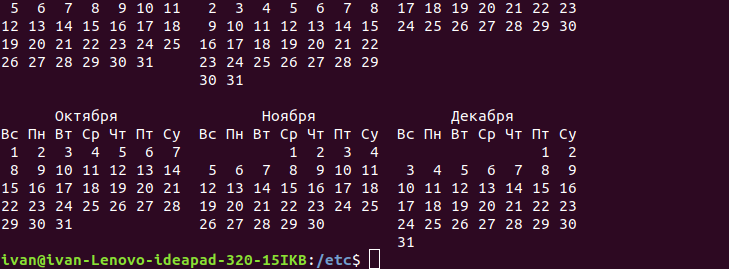
4. Виведіть на екран вміст відповідного файлу.



5. Виведіть календар на 1996 рік. 



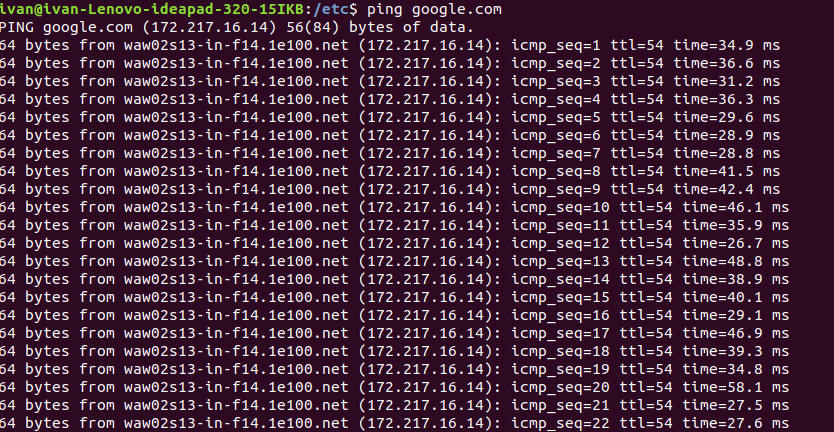
6. Виведіть календар на 1752 рік. Чи не помічаєте що-небудь цікаве у вересні?

7. Визначте, хто ще завантажений у систему.



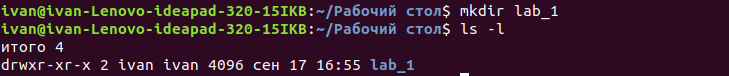
8. Наберіть команду ping. Поясніть результат.



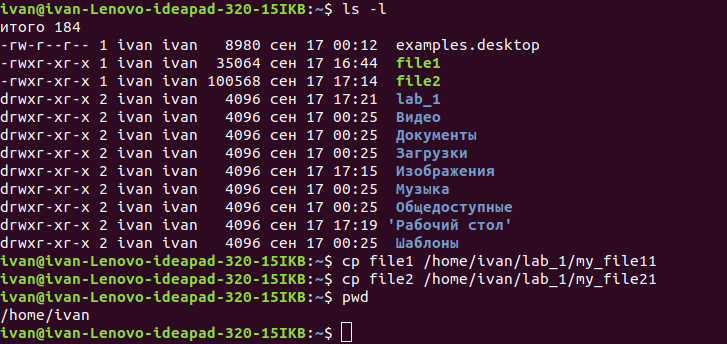
9.Скопіюйте файли.



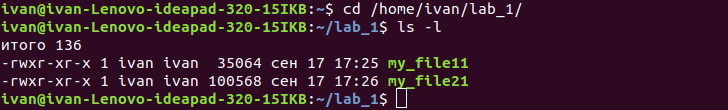
10.Створіть каталог lab\_1.



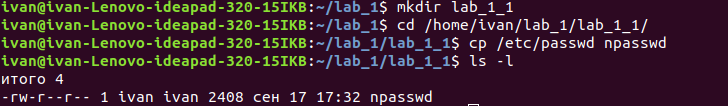
11. Скопіюйте в нього з вашого домашнього каталогу копію файлу 1, яку ви отримали в п.10, під ім'ям my\_ <ім’я файлу 1> . Перемістіть в цей каталог з вашого домашнього каталогу копію файлу 2 , яку ви отримали в п.10, перейменувавши його при цьому в my\_ <ім’я вихідного файлу 2> .



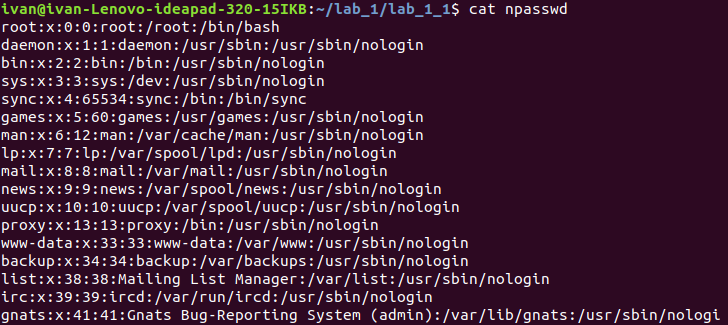
12.Перейдіть у свій домашній каталог і переконайтеся в тому, що все зроблено правильно.

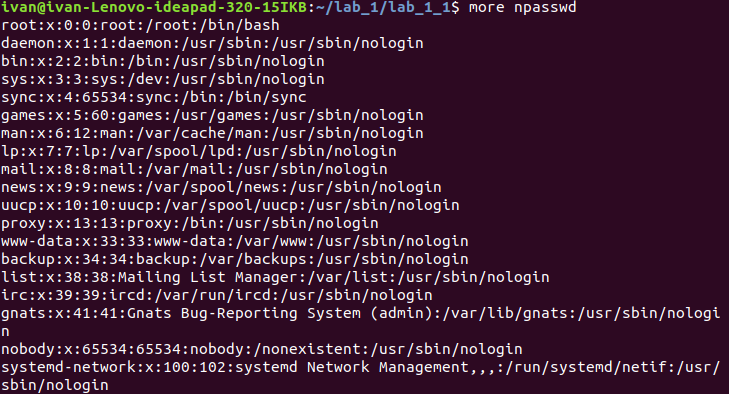


13.Створіть каталог lab\_1\_1 і перейдіть в нього. Скопіюйте в каталог lab\_1\_1 файл під ім'ям n<ім’я вихідного файлу>.

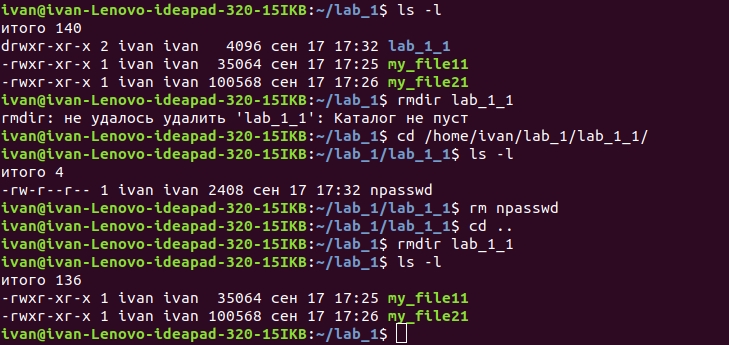


14.За допомогою команд cat і more перегляньте його вміст.



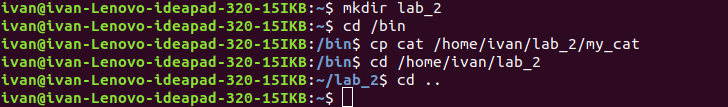


15.Видаліть каталог lab\_1\_1.

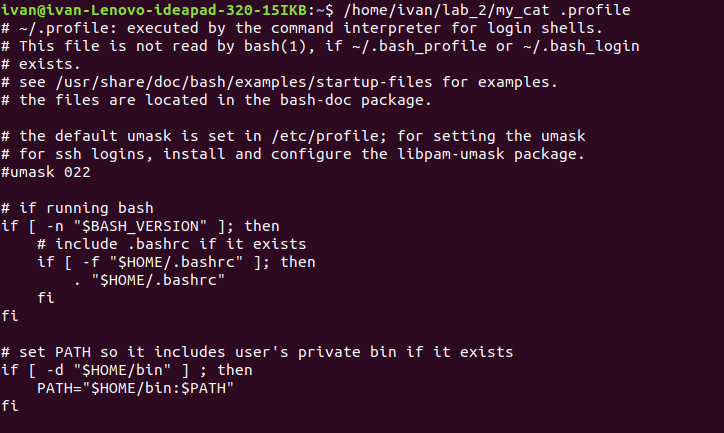


**Частина №2**

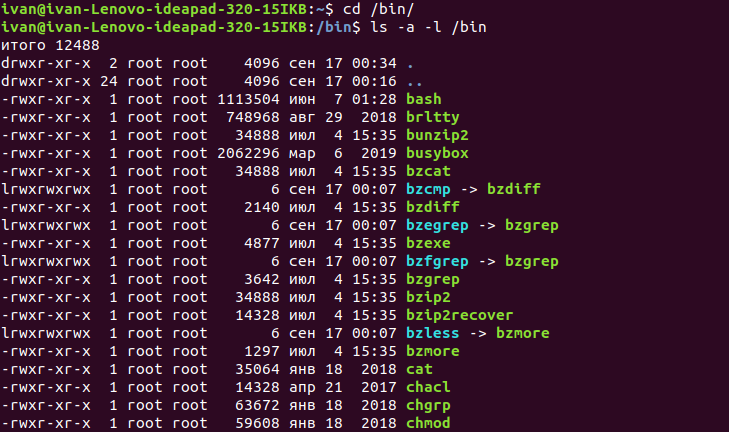
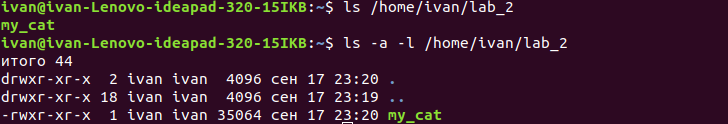
1. Створіть каталог lab\_2 та скопіюйте в каталог lab\_2 файл /bin/cat під назвою my\_cat.

****

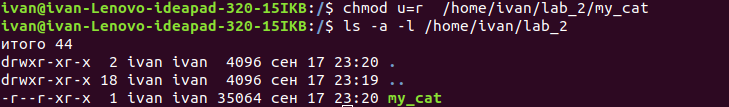
2. За допомогою файлу my\_cat, що знаходиться в каталозі lab\_2, перегляньте уміст файлу .profile (Ви знаходитесь у домашньому каталозі).



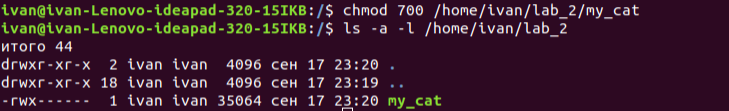
3. Перегляньте список файлів у каталозі lab\_2. Потім перегляньте список усіх файлів, включаючи приховані, з повною інформацією про файли. Зверніть увагу на права доступу, власника, дату модифікації файлу, що ви тільки-но скопіювали. Потім перегляньте цю інформацію про оригінальний файл (той, який копіювали) і порівняйте два результати.



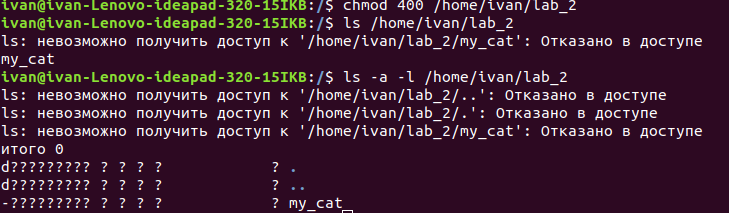
4. Змініть права доступу до файлу my\_cat так, щоб власник міг тільки читати цей файл.



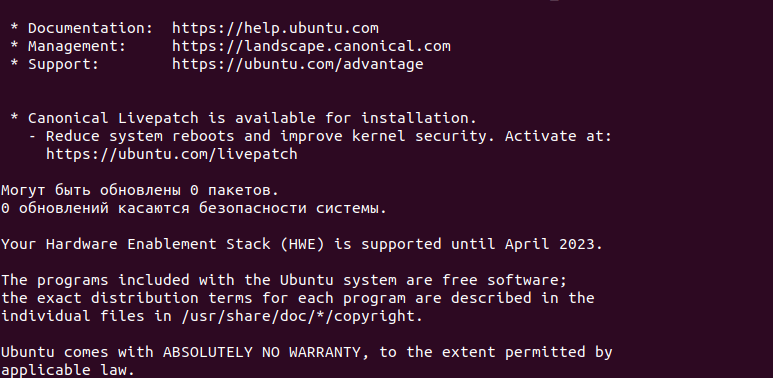
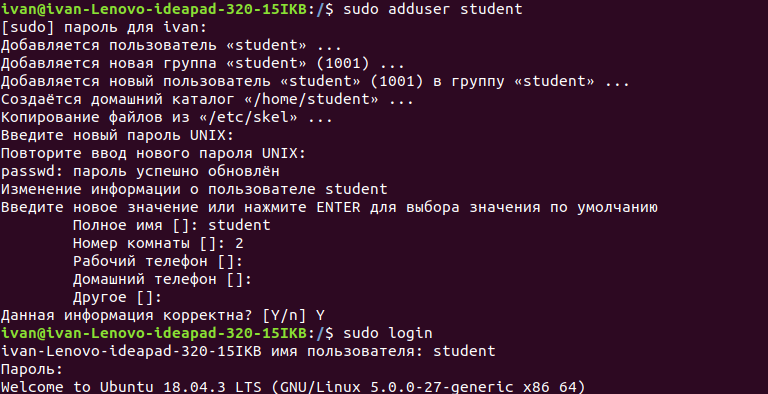
5. Визначте права на файл my\_cat таким чином, щоб Ви могли робити з файлом усе, що завгодно, а всі інші — нічого не могли робити.

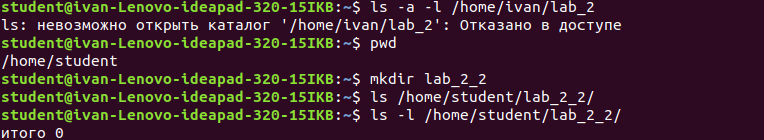


6. Поверніться в домашній каталог. Змініть права доступу до каталогу lab\_2 так, щоб ви могли його тільки читати. Спробуйте переглянути простий список файлів у цьому каталозі. Спробуйте переглянути список файлів з повною інформацією про них. Спробуйте запустити і видалити файл my\_cat з цього каталогу.

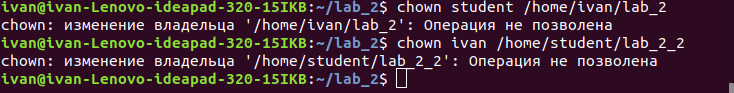


7.За допомогою команди su <user name> , завантажтесь в систему, користуючись обліковим записом іншого користувача. (Вам потрібно знати пароль цього користувача.) Спробуйте отримати доступ до Вашого каталогу lab\_2. Перевірте, чи правильно зроблено завдання попереднього пункту. Створіть каталог lab\_2\_2.

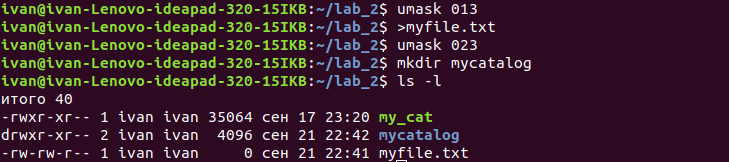




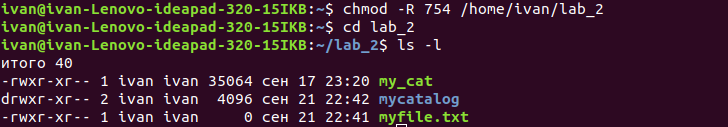
8.Знову завантажтесь в систему, користуючись своїм обліковим записом4 . Спробуйте зробити власником каталогу lab\_2 іншого користувача. Спробуйте зробити себе власником каталогу lab\_2\_2. Поясніть результати.



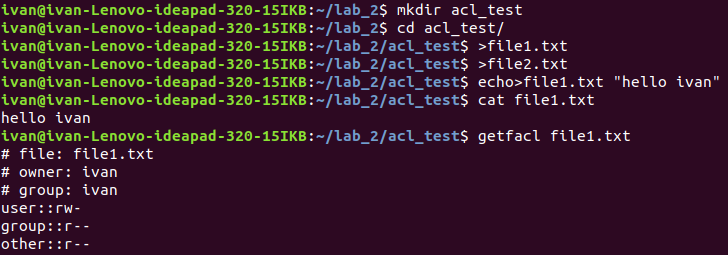
9.Зайдіть у каталог lab\_2. Зробіть так, щоб нові створені файли і каталоги діставали права доступу згідно Таблиці (див. наступну сторінку). Створіть новий файл і каталог і переконайтеся в правильності ваших установок.



10.Поверніть собі права читати, писати, та переглядати зміст каталогів.



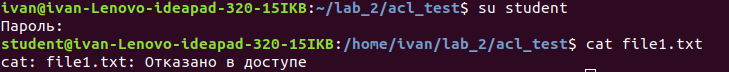
11.Створіть у каталозі lab\_2 каталог acl\_test та у ньому файли file1, file2. Під час створення file1 командою echo додайте до нього довільний текст. Виведіть ACL для file1



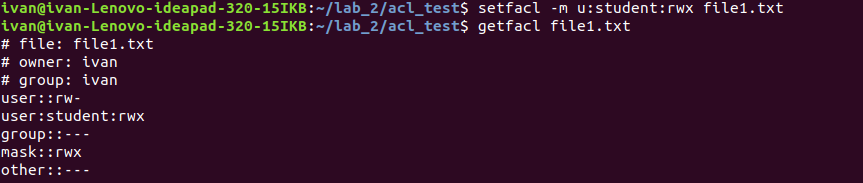
12.Змінить права доступу на file1 так, щоб тільки власник мав право на читання.



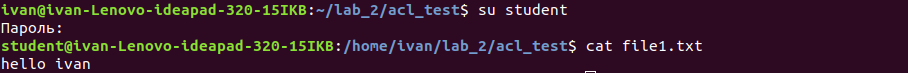
13.Увійдіть до системи під іншим обліковим записом та спробуйте прочитати вміст file1. Що отримаємо? Поверніться до свого облікового запису.



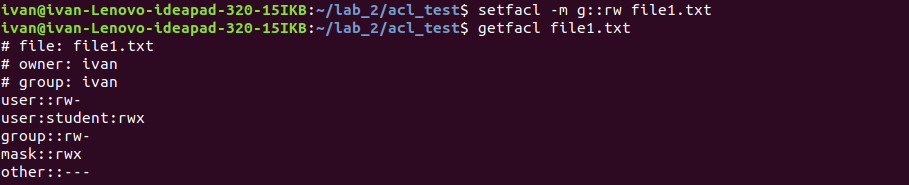
14.За допомогою команди setfacl додайте право на читання іншому обраному користувачу для file1. Перевірте, що створилось нове ACL для file1.



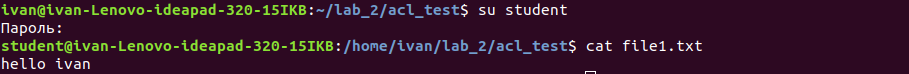
15.Увійдіть до системи під іншим обліковим записом та спробуйте прочитати вміст file1. Що отримаємо? Поверніться до свого облікового запису.



16.За допомогою команди setfacl встановіть значення маски таким чином щоб дозволити читати зміст file1 іншому користувачу. Виведіть ACL для file1



17.Увійдіть до системи під іншим обліковим записом, та спробуйте прочитати вміст file1. Ви повинні мати таку змогу.



**Висновок:** Виконуючи ці лабораторні роботи я здобув практичні навички в користуванні терміналом в операційній системі Linux. За допомогою команд login та passwd я входив в свій обліковий запис та змінював пароль також слід зазначити що в ОС Linux нового користувача можна додати прямо в терміналі через команду adduser. Звернемо увагу на те що копіювати, створювати, переміщати, видаляти можна не тільки “традиційним” способом але і за допомогою терміналу та набору таких команд як : cp, mv , rm, mkdir. Ще однією цікавою фішкою терміналу є те що ми можемо керувати доступом до файлів та каталогів , дозволяти та блокувати доступ різним користувачам та групам користувачів. Зауважимо що “гнучкість” настройки доступу пояснюється не тільки параметрами r, w, x але і широким набором функцій chmod, chown, getfacl, setfacl.