Міністерство освіти і науки України

НТУУ «КПІ ім. Ігоря Сікорського»

Навчально-науковий інститут атомної та теплової енергетики

Кафедра цифрових технологій в енергетиці

Лабораторна робота №5 з дисципліни «Операційна система UNIX»

**Створення сценаріїв в оболонці Bash**

**Варіант: 16**

Виконав студент групи ТР-12

Каркушевський Владислав

Перевірила д.т.н., проф. Левченко Л. О.

КИЇВ 2022

**Мета роботи :** набути навичок створювання bash-скриптів в ОС Linux.

**Теоретична частина**:

Командна оболонка BASH дозволяє створювати скрипти за допомогою групування декількох команд, які виконують певну дію. Скрипт - це звичайний текстовий файл, що містить системні або вбудовані команди оболонки. Такий файл може бути запущений на виконання наступним чином: $bash ім’я\_файла.

Оболонка послідовно інтерпретує і виконує команди, задані в сценарії. Ці ж команди можуть бути виконані простим послідовним викликом їх в командному рядку оболонки. Для файлів сценаріїв оболонки bash прийнято встановлювати розширення .sh. Тобто, для виконання скрипту необхідно запустити командну оболонку, передавши їй як параметр ім’я відповідного файлу. Є другий варіант запуску скрипту - вказати його ім'я в командній оболонці (тобто зробити з нього якийсь вид програми). Для цього треба в параметрах доступу визначити файл як виконуваний, і в перших рядках цього файлу явно вказати оболонку, для якої призначений цей скрипт, в такий спосіб: #!оболонка

Будь-який сценарій для Bash починається з вказівки в першому рядку комбінації: #!/bin/bash

Символи #! повідомляють системі про те, що наступний за ними аргумент – це програма, яка застосовується для виконання даного файлу. В даному випадку програма /bin/sh - командна оболонка, що застосовується за замовчуванням. Ця послідовність вказує на програму, яка використовується для обробки сценарію в командну оболонку bash. У загальному випадку символ «#» в скрипті означає коментар, що означає ігнорування рядка. Але якщо він є першим символом файлу і за ним слідує символ «!» та шлях до файлу (наприклад, /bin/bash, /bin/perl, /bin/ sh і т.д), командна оболонка запускає відповідний файл і передає йому його ім'я в якості аргументу.

**Створення сценарію**

Створити файл, який містить команди, можна за допомогою будь-якого текстового редактора.

**Перетворення сценарію** **у виконуваний файл**

Файл сценарію можна виконати двома способами.

Перший - запустити оболонку з ім'ям файлу сценарію як параметром: $ /bin/sh first. Цей варіант працює.

Другий спосіб (більш доцільний) - запускати сценарій, ввівши його ім'я і тим самим присвоївши йому статус інших команд Linux. Зробити це можна за допомогою команди chmod, змінивши режим файлу (file mode) і зробивши його виконуваним для всіх користувачів:

$ chmod u+x <ім’я сценарію>

Додані режими: для власника (User), виконання (x - eXecutable).

Аналогічну функціональність реалізує наступна команда:

chmod 744 <ім’я\_файлу\_сценарію>

Запуск сценарію на виконання з поточного каталогу проводиться за допомогою наступної команди: ./ <ім’я сценарію>

Зазначення шляху, який починається з символів «./», дає ще одну перевагу - ви випадково не зможете виконати іншу команду з тим же ім'ям, що і у вашого файлу сценарію. Після того як ви переконаєтеся в коректній роботі вашого сценарію, можете перемістити його в більш відповідне місце, ніж поточний каталог.

**Відладка сценаріїв**

При виявленні помилки при виконанні сценарію командна оболонка виводить на екран номер рядка, що містить помилку. Якщо помилку відразу не видно, потрібно додати кілька додаткових команд echo для виведення значень змінних, протестувати фрагменти програмного коду, вводячи їх в командній оболонці в інтерактивному режимі. Основний спосіб відстеження помилок, які найбільш складно виявляються - використання опцій відладки командної оболонки. Опції відладки командного рядка:

Опція Призначення

sh -n Тільки перевіряє синтаксичні помилки

sh -v Виводить на екран команди перед їх виконанням

sh -x Виводить на екран команди після обробки командного рядка

sh -u Видає повідомлення про помилку при використанні невизначеної змінної

**Завдання:**

1. Ознайомитися з теоретичними матеріалом по лабораторній роботі.

2. Опанувати команди, які використовують при написанні сценарію.

3. Підготувати звіт для викладача про виконання лабораторної роботи і представити його.

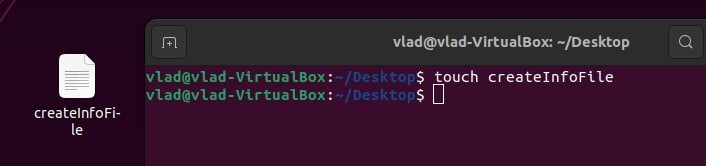
Розробити скрипт, який виконує зазначені дії згідно варіанту по списку журналу групи. Створити файл з назвою власного прізвища, записати до нього ПІБ студента, групу навчання, за бажанням додаткову інформацію щодо ваших уподобань. Створити скрипт, який виводить ПІБ студента, групу навчання, коментарі щодо майбутніх дій, необхідно робити затримку після видачі результатів на екран. Скрипт повинен містити функції, які виконують зазначені дії, а також додатково записати дію відповідно до варіанту.

**Завдання за варіантом:**

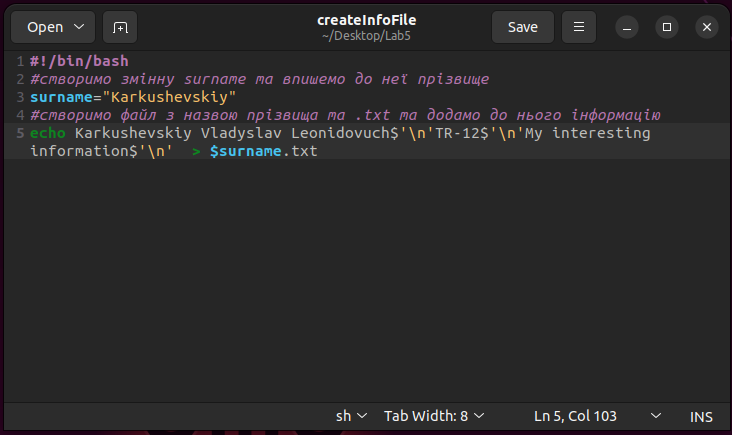
16.Вивести версію системи.

**Хід виконання роботи**

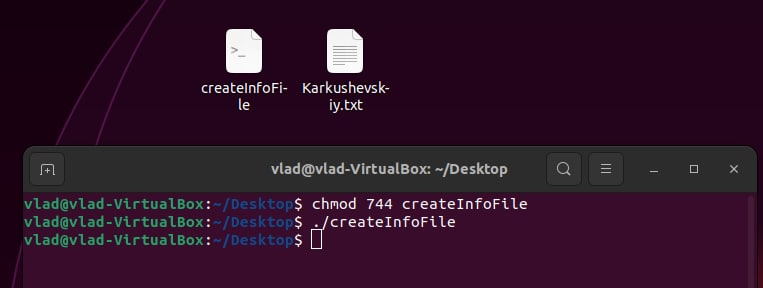
Першим ділом, у консолі, створимо текстовий файл **createInfoFile,** у який будемо записувати наший скрипт. Робимо це за допомогою команди **touch,** і у результаті створюється файл на робочому столі



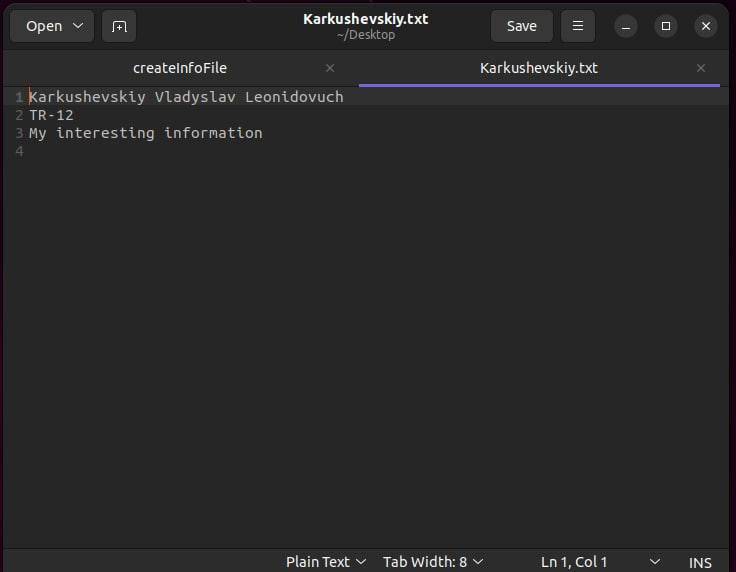
Після створення, переходимо у цей файл, та змінюємо його тип з текстового, в **sh.** Для файлів сценаріїв оболонки bash прийнято встановлювати розширення .sh. У середині файлу у першому рядку прописуємо **#!/bin/bash** для того, аби вказати що це програма, яка застосовується для виконання даного файлу. Далі створюємо змінну **surname** у яку вписуємо своє прізвище. Після цього створюємо текстовий файл зі змістом змінної surname, та заносимо інформацію у цей файл.



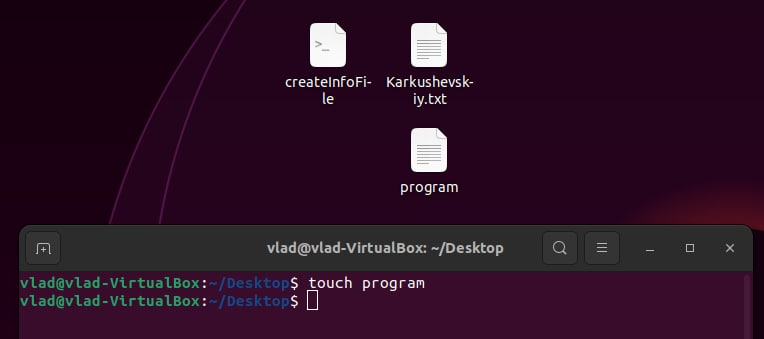
Далі, наступним кроком, перетворюємо сценарій у виконуваний файл, за допомогою команди **chmod 744.** Після цього запускаємо виконуваний файл за допомогою **./**  У результаті, на робочому столі, створюється файл **Karkushevskiy.txt**



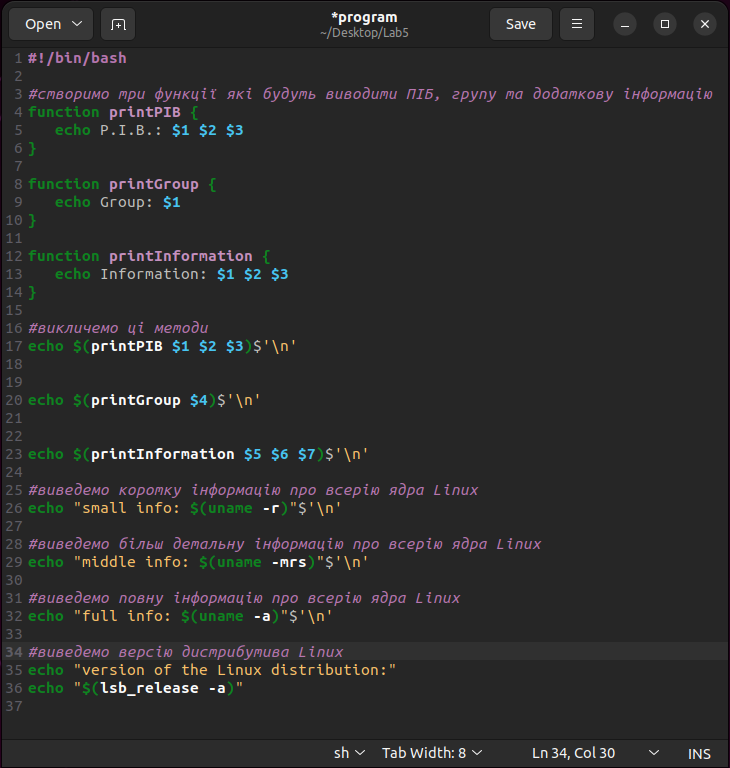
Зміст файлу **Karkushevskiy.txt** виглядає наступним чином:



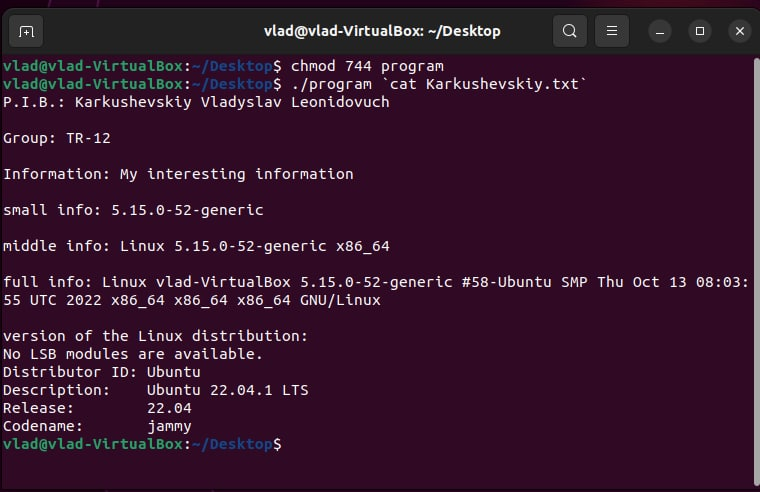
Після цього створимо новий текстовий файл **program**, у якому буде зберігатися інший скрипт, який буде виводити інформацію та виконувати завдання за варіантом.



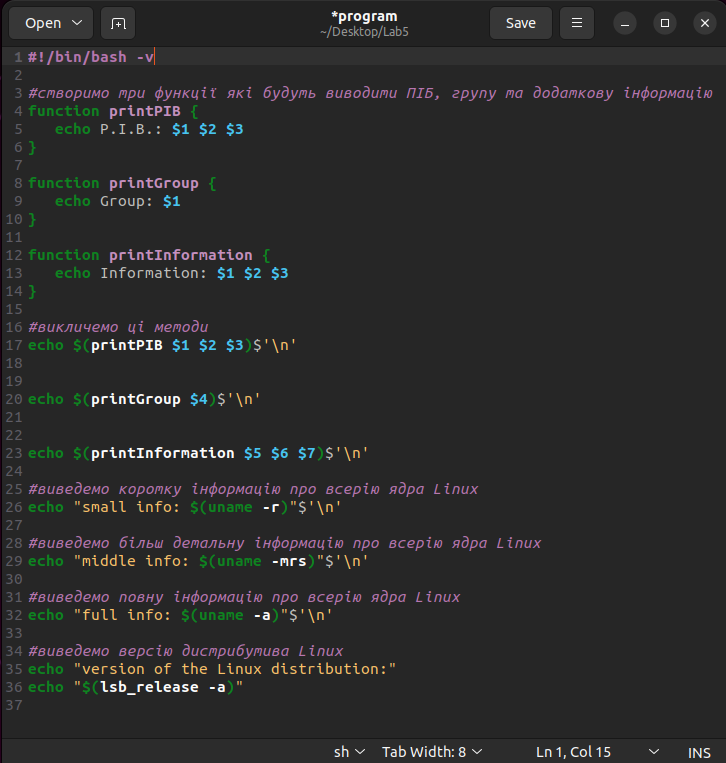
Далі запишемо команди до скрипту. У першому рядку так само вкажемо **#!/bin/bash** . Далі створимо три функції: printPIB, printGroup, printInformation. Ці функції будуть виводити інформацію, яка буде передаватися у параметр. Після функції, викликаємо їх, та передаємо у параметр відповідні значення які будуть передавати у параметр скрипту. Тоді виконуємо завдання згідно варіанту, та виводимо коротку, середню, та повну інформацію про версію системи.



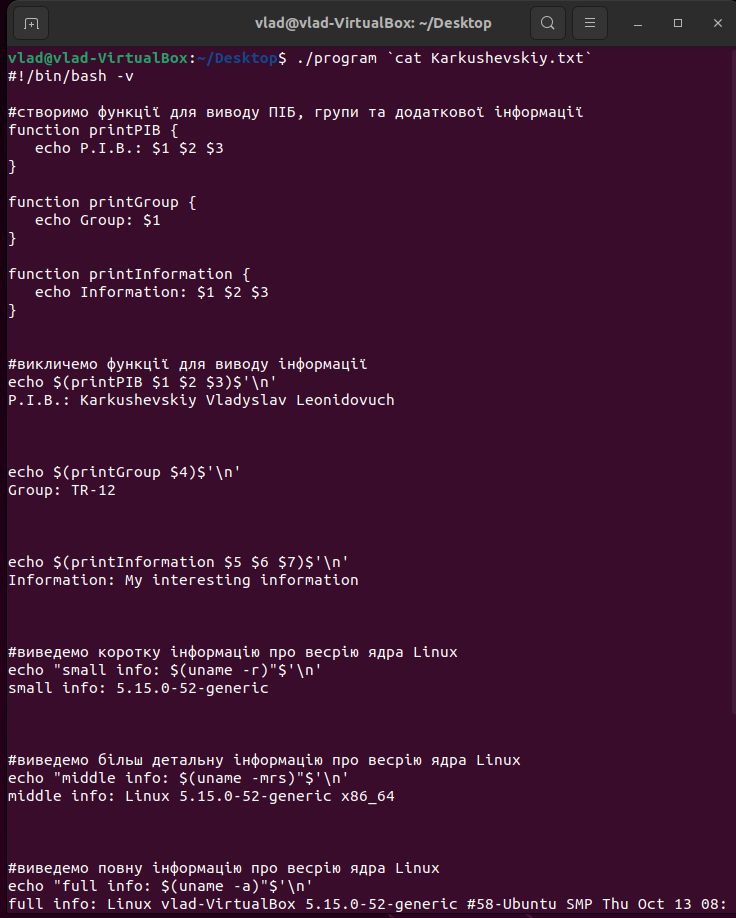
Далі, наступним кроком, перетворюємо сценарій у виконуваний файл, за допомогою команди **chmod 744.** Після цього запускаємо виконуваний файл за допомогою **./** та передаємо у параметр зміст файлу **Karkushevskiy.txt**. Після чого у консолі виводиться уся інформація



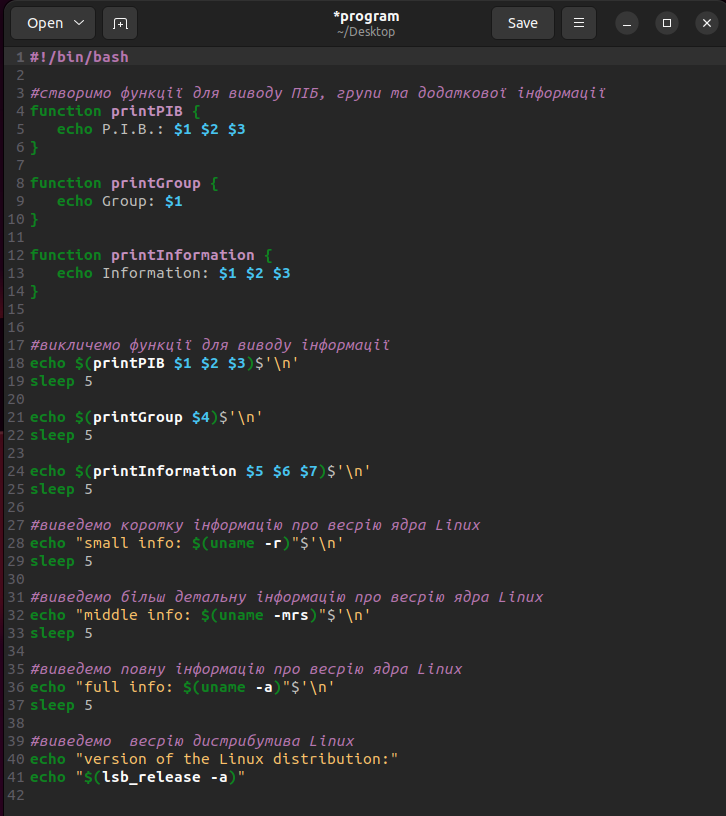
Також можемо використати опції відладки сценарію. Наприклад використаємо опцію sh -v яка виводить на екран команди перед їх виконанням. Пропишемо її у першому рядку



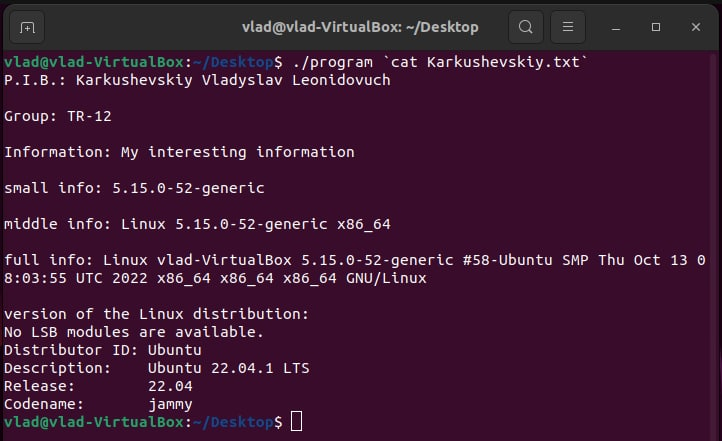
Після цього, виконавши скрипт результат буде виглядати наступним чином:



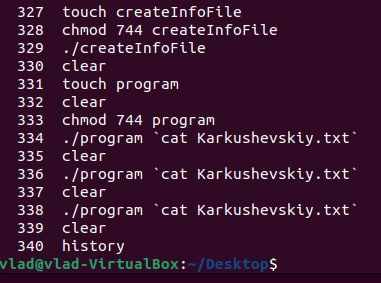
Також створимо затримку після видачі результатів на екран. Використаємо команду **sleep** та вкажемо значення 5, що означає, що результат буде виводитися після 5-секундної паузи.



Результат буде той самий, але інформація буде виводитися з 5-секундною паузою



Історія команд виглядає так:



**Висновок:** На цій лабораторній роботі було ознайомлено з теорію по лабораторній роботі. Набуто навички створювання bash-скриптів в ОС Linux. Ознайомлено з створенням сценаріїв, перетворення сценарію у виконуваний файл, синтаксисом команд для створення сценарію та відладкою сценарію. Створено два скрипта та виконано завдання згідно варіанту.