“Київський фаховий коледж зв’язку”

Циклова комісія Комп’ютерної інженерії

ЗВІТ ПО ВИКОНАННЮ

ЛАБОРАТОРНОЇ РОБОТИ №7

з дисципліни: «Операційні системи»

Тема: «Створення скриптових сценаріїв та визначення апаратної конфігурації системи»

Виконала

студентка

групи БІКС-13

Лось Злата Володимирівна

Перевірив викладач

Сушанова В.С.

Київ 2024

**Мета роботи:**

1. Отримання практичних навиків роботи з командною оболонкою Bash.
2. Знайомство знайомство з базовими діями при роботі зі скриптовими сценаріями.

**Матеріальне забезпечення занять:**

1. ЕОМ типу IBM PC.

2. ОС сімейства Windows та віртуальна машина Virtual Box (Oracle).

3. ОС GNU/Linux (будь-який дистрибутив).

4. Сайт мережевої академії Cisco netacad.com та його онлайн курси по Linux.

**Завдання для попередньої підготовки:**

1. \*Прочитайте короткі теоретичні відомості до лабораторної роботи та зробіть невеликий словник базових англійських термінів з питань призначення команд та їх параметрів.

Shell Script - це текстовий файл, що містить низку команд, які можуть бути виконані оболонкою або базовою операційною системою.

Shebang - послідовність із двох символів: «решітки» та знака оклику (#!), яка ставиться на початку файлу (текстового), написаного на скриптовій мові.

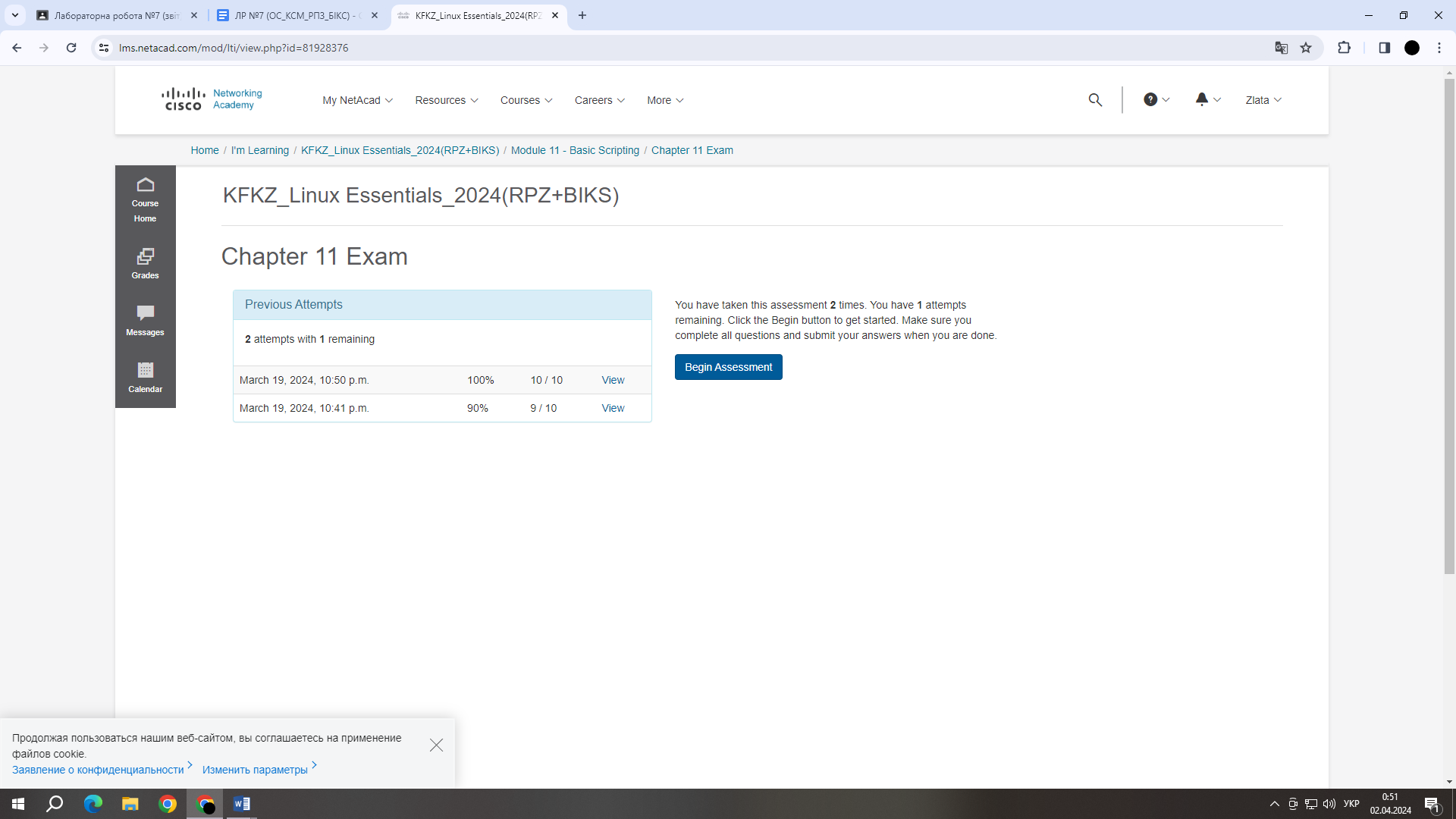
Nano — текстовий редактор для консолі.

2. Вивчіть матеріали онлайн-курсу академії Cisco “NDG Linux Essentials”:

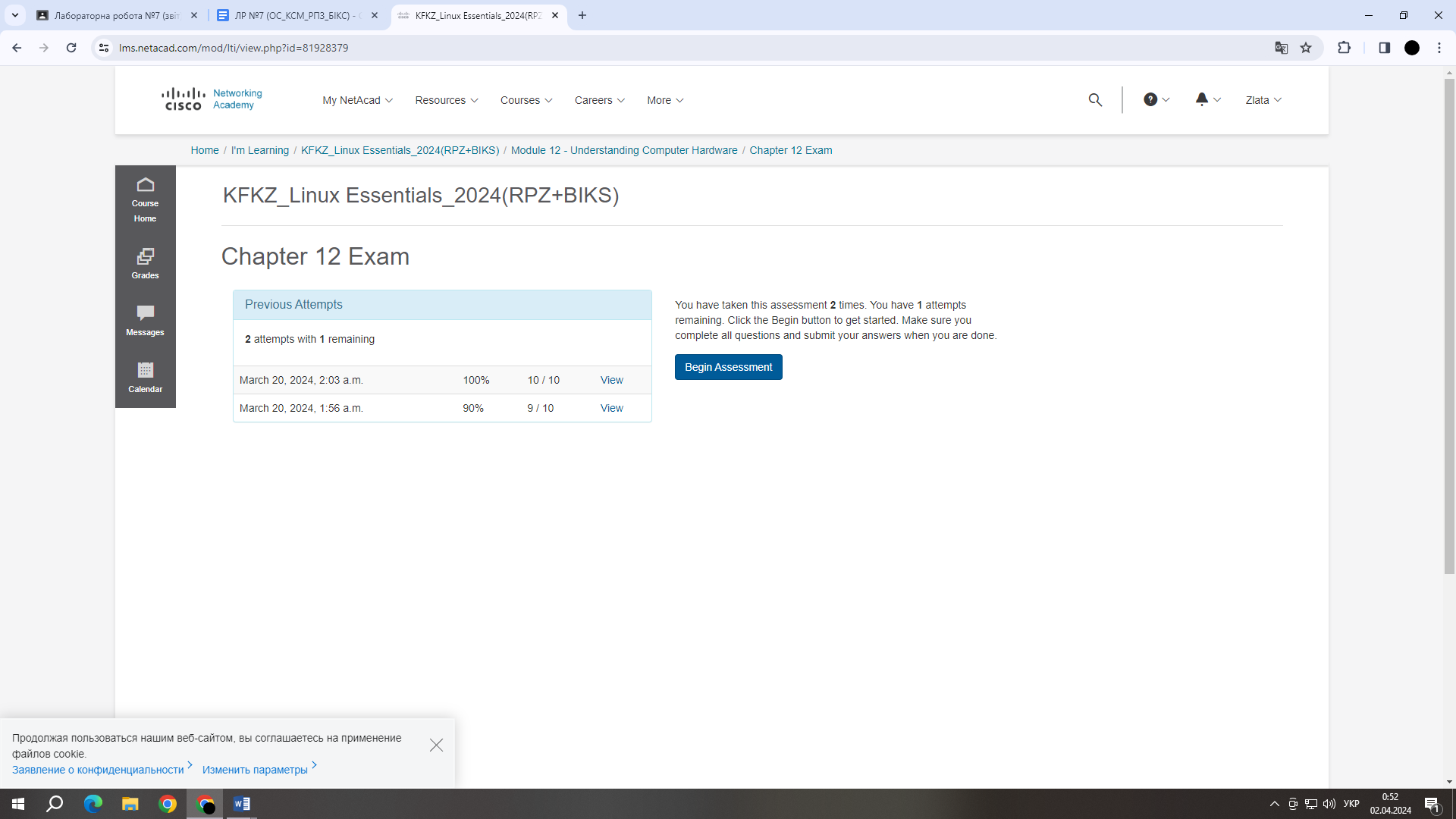
* Chapter 11 - Basic Scripting
* Chapter 12 - Understanding Computer Hardware

3. Пройдіть тестування у курсі NDG Linux Essentials за такими темами:

* Chapter 11 Exam



* Chapter 12 Exam



4. На базі розглянутого матеріалу дайте відповіді на наступні питання:

**4.1. \*Охарактеризуйте поняття скриптового сценарію у командній оболонці.**

Скриптовий сценарій у командній оболонці - це файл, який містить послідовність команд, що виконуються в командному рядку операційної системи. Ці скрипти дозволяють автоматизувати повторювані завдання шляхом виконання декількох команд послідовно або в залежності від умов.

**Основні характеристики скриптових сценаріїв у командній оболонці:**

1. Мова програмування: Скриптові сценарії можуть бути написані на різних мовах програмування, таких як Bash, Python, Perl, або інших.

2. Виконання команд: Вони містять послідовність команд, які виконуються автоматично.

3. Змінні та умови: Сценарії можуть містити змінні, умови, цикли та інші конструкції програмування для керування потоком виконання.

4. Параметризація: Сценарії можуть приймати аргументи з командного рядка, що дозволяє їх використовувати з різними вхідними даними.

5. Автоматизація задач: Вони дозволяють автоматизувати повторювані завдання, що збільшує ефективність та знижує ймовірність помилок.

6. Підтримка функцій: Сценарії можуть містити функції, що спрощує структуру коду та полегшує його розуміння та модифікацію.

7. Вивід інформації: Сценарії можуть виводити інформацію на екран, що дозволяє контролювати та відстежувати процес виконання.

**4.2. \*Яким чином створюються та редагуються скрипти, що треба зробити щоб запустити скрипт?**

Щоб створити та редагувати скрипти, вам зазвичай знадобиться текстовий редактор, такий як Notepad (для Windows), Nano, Vim, або Emacs (для Unix-подібних систем). Ось кроки для створення та редагування скриптів:

1. Запустіть текстовий редактор на вашому комп'ютері.

2. Напишіть код скрипта, використовуючи синтаксис мови програмування, на якій ви плануєте писати скрипт.

3. Збережіть файл з іменем, яке ви виберете, і розширенням, яке вказує на мову програмування. Наприклад, якщо ви пишете скрипт на Bash, ім'я файлу може бути myscript.sh.

Щоб запустити скрипт:

1. Відкрийте термінал або командний рядок на вашому комп'ютері.

2. Використовуйте команду cd, щоб перейти до каталогу, де зберігається ваш скрипт.

3. Виконайте ваш скрипт, введіть його ім'я у командний рядок і натисніть Enter. Наприклад, для скрипта myscript.sh ви можете виконати команду ./myscript.sh.

4. В Windows вам може знадобитися запустити скрипт через командний рядок або подвійним клацанням по файлу скрипта (якщо він налаштований для цього).

**4.3. \*\*Які основні компоненти материнської плати ви знаєте?**

* Сокет процесора: Роз'єм, в якому встановлюється процесор. Це дозволяє процесору взаємодіяти з іншими компонентами системи.
* Чіпсет: Це набір мікросхем на материнській платі, які керують роботою інших компонентів, таких як швидкість передачі даних між процесором, оперативною пам'яттю та іншими пристроями.
* Слоти пам'яті (DIMM/RAM): Слоти, в які встановлюється оперативна пам'ять (RAM). Вони забезпечують доступ процесора до пам'яті для зберігання тимчасової інформації під час роботи системи.
* Слоти розширення (PCIe/PCI/AGP): Слоти, які дозволяють встановлювати розширювальні картки, такі як відеокарти, звукові карти, мережеві адаптери та інші пристрої.
* З'єднувальні роз'єми (SATA/IDE): Роз'єми для підключення жорстких дисків, оптичних приводів та інших пристроїв зберігання даних до материнської плати.
* BIOS/UEFI чіп: Це програмне забезпечення, що забезпечує початкову ініціалізацію системи та управління різними функціями апаратного забезпечення.
* Роз'єм живлення (ATX/EPS): Роз'єм для підключення живлення від блока живлення до материнської плати.
* Зовнішні порти (USB/Ethernet/Audio): Роз'єми для підключення зовнішніх пристроїв, таких як клавіатура, миша, монітор, мережевий кабель, аудіо пристрої та інші.
* Кнопки і світлодіоди: Наприклад, кнопка включення, кнопки скидання BIOS, LED-індикатори статусу живлення, активності диска та інші.

**4.4. \*\*Коротко охарактеризуйте для яких пристроїв оперують поняттями MBR та GPT?**

**1. MBR (Master Boot Record):**

* Використовується для BIOS-сумісних комп'ютерів та операційних систем.
* Підтримує обсяги до 2 терабайт.
* Дозволяє до чотирьох основних розділів (primary partitions), або трьох основних розділів та одного розширеного розділу (extended partition), який може містити кілька логічних розділів.
* Широко підтримується операційними системами, такими як Windows, Linux та багатьма іншими.

**2. GPT (GUID Partition Table):**

* Використовується для UEFI-сумісних комп'ютерів та операційних систем.
* Підтримує великі обсяги даних (більше 2 терабайт).
* Дозволяє до 128 основних розділів (primary partitions).
* Містить більш детальну інформацію про розділи та має покращені можливості резервування та відновлення даних.
* Частіше використовується для сучасних версій операційних систем, таких як Windows 10, macOS, Linux і багато інших.

**4.5. \*\*В чому суть операції монтування, для чого вона потрібна?**

Суть операції монтування полягає в тому, що вона дозволяє операційній системі доступатися до даних, які знаходяться на зовнішніх носіях або інших пристроях. Оскільки операційна система організована відповідно до ієрархічної структури файлової системи, монтування дозволяє підключити зовнішні пристрої до цієї ієрархії, надаючи їм шлях і точку доступу до даних.

5. Підготувати в електронному вигляді початковий варіант звіту:

- Титульний аркуш, тема та мета роботи

- Словник термінів

- Відповіді на п.4.1 та п.4.5 з завдань для попередньої підготовки

**Хід роботи:**

1.Початкова робота в CLI-режимі в Linux ОС сімейства Linux:

1.1. Запустіть віртуальну машину VirtualBox, оберіть CentOS та запустіть її. Виконайте вхід в систему під користувачем: CentOS, пароль для входу: reverse (якщо виконуєте ЛР у 401 ауд.) та запустіть термінал.

1.2. Запустіть віртуальну машину Ubuntu\_PC (якщо виконуєте завдання ЛР через академію netacad)

1.3. Запустіть свою операційну систему сімейства Linux (якщо працюєте на власному ПК та її встановили) та запустіть термінал.

2. Опрацюйте всі приклади команд, що представлені у лабораторних роботах курсу NDG Linux Essentials - Lab 11: Basic Scripting та Lab 12: Understanding Computer Hardware. Створіть таблицю для опису цих команд.

|  |  |
| --- | --- |
| Назва команди | Її призначення та функціональність |
| lscpu | використовується для виведення інформації про процесор (CPU) системи |
| lspci | використовується для виведення інформації про всі PCI (Peripheral Component Interconnect) пристрої, що підключені до системи через шину PCI |
| lsusb | використовується для виведення інформації про всі USB-пристрої, підключені до системи |
| lsmod | використовується для виведення списку завантажених модулів ядра (kernel modules) |
| fdisk | використовується для роботи з розділами жорсткого диска |

3. Створіть скриптові сценарії з виводом текстових повідомлень для користувача (продемонструйте скріншоти):

- сценарій має виводити привітання до поточного користувача вказуючи поточну дату та інформацію про поточну систему;

- \*сценарій має виводити інформацію про апаратну конфігурацію поточної системи (використовуйте команди розглянуті в Lab 12: Understanding Computer Hardware);

- \*\*наведіть свій приклад скриптового сценарію.

**Контрольні запитання:**

**1. В чому відмінність між командами arch та lscpu?**

Відмінність полягає у тому, що команда **arch** надає загальну інформацію про архітектуру операційної системи, тоді як команда **lscpu** специфічно вказує на деталі процесора, які можуть бути корисними для аналізу апаратного забезпечення системи.

**2. Якою командою можна отримати інформацію про стан використання RAM поточною системою?**

Інформацію про стан використання оперативної пам'яті (RAM) поточною системою можна отримати за допомогою команди `free`.

Команда `free` виводить статистику використання пам'яті у системі, включаючи загальний обсяг доступної пам'яті, використану пам'ять, вільну пам'ять, використання подвійного обміну (swap), а також інші важливі показники.

**3. \*Яким чином у скриптах можна опрацьовувати змінні та створювати розгалужені та циклічні сценарії?**

1. Змінні:

* Встановлення значення змінної: variable=value.
* Використання значення змінної: echo $variable.
* Параметри командного рядка: $1, $2, і так далі, щоб отримати значення переданих аргументів.

2. Розгалужені сценарії:

* Умовні конструкції: if, elif, else.

3. Циклічні сценарії:

* Цикли for, while.

**4. \*Які команди для перегляду стану підключення периферійних пристроїв можна використати в терміналі?**

* **lsusb:** Ця команда виводить список всіх USB-пристроїв, підключених до вашої системи. Вона надає інформацію про виробника, модель та інші деталі про кожен підключений пристрій USB.
* **lspci:** Ця команда виводить список всіх PCI-пристроїв, підключених до вашої системи. Вона надає детальну інформацію про графічні карти, мережеві адаптери, аудіо- та відеоадаптери та інші PCI-пристрої.
* **lsblk:** Ця команда виводить список всіх блочних пристроїв, таких як жорсткі диски та флеш-накопичувачі, які підключені до вашої системи. Вона також показує інформацію про їхні розділи.
* **lscpu:** Ця команда виводить інформацію про ваш процесор, включаючи кількість ядер, модель, частоту та інші характеристики.

**5. \*\*Які можливості застосунку gparted?**

* **Створення, видалення та редагування розділів:** GParted дозволяє створювати нові розділи на жорсткому диску, видаляти існуючі розділи та змінювати їхні параметри, такі як розмір, файлову систему та тип.
* **Форматування розділів:** Ви можете форматувати розділи з різними файловими системами, такими як ext4, NTFS, FAT32 та інші, використовуючи GParted.
* **Перейменування розділів**: За допомогою GParted ви можете перейменувати розділи на вашому жорсткому диску для зручності.
* **Клонування розділів:** GParted дозволяє створювати копії розділів та виконувати їх клонування, включаючи операції копіювання між різними дисками або розділами.
* **Переміщення розділів:** Ви можете переміщати розділи на вашому жорсткому диску без втрати даних, розширювати або скорочувати їх.
* **Перевірка та відновлення файлових систем: GParted** має можливість виконувати перевірку та відновлення різних типів файлових систем, що може допомогти у виявленні та виправленні проблем з файловою системою.
* **Робота з RAID та LVM:** GParted підтримує операції зі створенням, видаленням та редагуванням розділів у конфігураціях RAID та LVM.

**Висновок:** Отримала практичні навички роботи з командною оболонкою Bash. Ознайомилась з базовими діями при роботі зі скриптовими сценаріями.