Kyiv College of Communications

PERFORMANCE REPORT

Laboratory work No. 7

Discipline: "Operating Systems"

Topic: “Creating script scenarios and determining the system hardware configuration”

Performed by students of Group CSN-33 *(Computer Systems and Networks)*.:

Kuznetsov Artur Serhiyovych

Finchuk Alina Oleksiivna

Checked by:

Sushanova V.S.

Kyiv – 2025

**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.

|  |  |  |
| --- | --- | --- |
| **English term** | **Ukrainian translation** | **Meaning / Пояснення** |
| **Shell** | Оболонка | Програма, що приймає команди користувача (bash, sh, zsh). |
| **Script** | Скрипт | Текстовий файл з послідовністю команд, які виконуються автоматично. |
| **Shebang (#!)** | Шебанг | Перша строка в скрипті, яка вказує інтерпретатор (наприклад: #!/bin/bash). |
| **Executable** | Виконуваний | Файл, який має дозвіл на запуск. |
| **Permission** | Права доступу | Визначають, хто може читати, змінювати чи запускати файл. |
| **Command** | Команда | Інструкція, яку оболонка виконує. |
| **Variable** | Змінна | Тимчасове сховище даних у скрипті. |
| **Loop** | Цикл | Повторення команд певну кількість разів. |
| **Condition** | Умова | Перевірка певного стану (if, then, else). |
| **Interpreter** | Інтерпретатор | Програма, що виконує скрипт (bash, python тощо). |
| **Mount** | Монтування | Підключення файлової системи до системного каталогу. |

1. На базі розглянутого матеріалу дайте відповіді на наступні питання:
   1. \*Охарактеризуйте поняття скриптового сценарію у командній оболонці.  
      Скриптовий сценарій (shell script) — це **текстовий файл, який містить послідовність команд**, що виконуються командною оболонкою (наприклад, Bash). Він використовується для **автоматизації рутинних дій**, наприклад: запуск системних перевірок, копіювання файлів, створення резервних копій тощо.  
       На початку скрипту зазвичай вказується рядок #!/bin/bash, який визначає, якою оболонкою його потрібно виконати.
   2. \*Яким чином створюються та редагуються скрипти, що треба зробити щоб запустити скрипт?

* **Створення скрипту:**  
   Використовують текстовий редактор, наприклад nano, vim або gedit. nano test.sh
* **Приклад скрипту:**

#!/bin/bash  
echo "Hello, $USER!"  
echo "Today is $(date)"

* **Надати права на виконання:** chmod +x test.sh
* **Запуск скрипту:** ./test.sh або bash test.sh
  1. \*\*Які основні компоненти материнської плати ви знаєте?

**CPU Socket** — роз’єм для центрального процесора.

**RAM Slots** — слоти для оперативної пам’яті.

**Chipset** — набір мікросхем, що керує взаємодією компонентів.

**BIOS/UEFI** — мікропрограма для початкового завантаження системи.

**Expansion Slots (PCIe)** — для підключення відеокарт, мережевих чи звукових карт.

**SATA / M.2 / NVMe порти** — для накопичувачів.

**Power Connectors** — для живлення.

**I/O Ports** — роз’єми для підключення зовнішніх пристроїв (USB, HDMI, Ethernet).

* 1. \*\*Коротко охарактеризуйте для яких пристроїв оперують поняттями MBR та GPT?
* **MBR (Master Boot Record)** — старий тип розмітки дисків, який підтримує до 4 розділів і диски до 2 ТБ.
* **GPT (GUID Partition Table)** — сучасний стандарт, що використовується з UEFI, підтримує великі об’єми дисків (до 9.4 ZB) і до 128 розділів.  
   Застосовуються для **жорстких дисків (HDD)** та **твердотільних накопичувачів (SSD)**.
  1. \*\*В чому суть операції монтування, для чого вона потрібна?

Монтування — це **процес підключення файлової системи пристрою** (наприклад, флешки або диска) до дерева каталогів Linux.  
 Без монтування ОС не “бачить” пристрій.  
 Приклад: sudo mount /dev/sdb1 /mnt/usb

Після виконання команда /mnt/usb стає точкою доступу до даних на флешці.

**Хід роботи:**

* 1. Початкова робота в CLI-режимі в Linux ОС сімейства Linux:
  2. Запустіть операційну систему Linux Ubuntu. Виконайте вхід в систему та запустіть термінал ***(якщо виконуєте ЛР у 401 ауд.)***..
  3. Запустіть віртуальну машину Ubuntu\_PC ***(якщо виконуєте завдання ЛР через академію netacad)***
  4. Запустіть свою операційну систему сімейства Linux ***(якщо працюєте на власному ПК та її встановили)*** та запустіть термінал.

1. 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 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

**Примітка:** **Скріншоти** виконання команд в терміналі можна **не представляти**, достатньо **коротко описати команди в таблиці**.

* 1. 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?**

Команда **arch** просто показує тип процесора або архітектуру системи, наприклад x86\_64.  
 А **lscpu** дає набагато детальнішу інформацію — модель процесора, кількість ядер, потоки, кеш, тощо.  
 Тобто arch — коротко, а lscpu — більш детально.

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

Для цього є команда **free**.  
 Якщо написати free -h, то покаже, скільки пам’яті всього, скільки зайнято, вільно, кеш і swap — і все у зручному вигляді.

### **3. Як у скриптах можна опрацьовувати змінні та створювати розгалужені й циклічні сценарії?** У скриптах можна створювати змінні просто так:

name="Alina"  
echo "Hello, $name"  
Для розгалуження використовують **if...then...else**, а для повторень — **for**, **while**.  
 Наприклад:

if [ $age -ge 18 ]; then  
 echo "Adult"  
else  
 echo "Minor"  
fi  
Це дозволяє скрипту реагувати по-різному залежно від умов.

### **4. Які команди для перегляду стану підключення периферійних пристроїв можна використати в терміналі?** Можна подивитись через:

* lsusb — показує USB-пристрої,
* lspci — пристрої, що підключені через PCI (відеокарта, мережева карта і т.д.),
* lsblk — диски й розділи,
* df -h — використання місця на носіях.  
   Цього зазвичай достатньо, щоб побачити, що підключено.

### **5. Які можливості застосунку gparted?** **GParted** — це зручна програма з графічним інтерфейсом для роботи з розділами диска. Через неї можна створювати, видаляти, змінювати розмір і форматувати розділи, а ще — дивитись тип розмітки (MBR або GPT). Тобто це як «диспетчер розділів», тільки для Linux.

**Conclusion Finchuk Alina**

**Conclusion by Kuznetsov Artur**