“Київський фаховий коледж зв’язку”

Циклова комісія Комп’ютерної інженерії

**ЗВІТ ПО ВИКОНАННЮ**

**ЛАБОРАТОРНОЇ РОБОТИ №6**

з дисципліни: «Операційні системи»

**Тема: “Створення скриптових сценаріїв та визначення апаратної конфігурації системи”**

Виконав студент

групи БІКС-13

Когут Ігор Святославович

Береза Артем Шихайович

Перевірив викладач

Сушанова В.С.

Київ 2022

**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 (Windows 7).**

**3. Віртуальна машина – Virtual Box (Oracle).**

**4. Операційна система GNU/Linux – CentOS.**

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

**Завдання для попередньої підготовки ( Когут )**

|  |  |
| --- | --- |
| **Термін англійською** | **Термін українською** |
| Shebang | Шебан |
| Hashbang line | Рядок shebang |
| Editor | Редактор |
| Nano | Нано |
| Variables | Змінні |
| Loops | Цикли |
| Positional parameter | Позиційний параметр |
| Named parameter | Іменований параметр |
| Conditionals | Умови |
| Option | Опція |

4) 4.1. A script script in a command shell (shell script) is a sequence of commands and instructions written in a text file, which are executed sequentially when this file is started. These scripts allow you to automate various tasks and actions in the command environment of the operating system.

Сharacteristics of script scripts:

1) They are written in text format, so they can be easily created and edited.

2)They can be executed at any time without the need for the user to directly enter commands.

3)They can be used to perform complex tasks that are difficult or impossible to perform manually.

4.2. Scripts are created and edited using a text editor. For this we can use

3 text editors of our choice, for example: Nano, Vim or Emacs.

In order for us to create a new script, we need to open a text editor and enter the sequence of commands to be executed (#! /bin/bash)

To run the script, you need to enter its name in the command line. For example, to run a script with a name, in our case it has the name: "hello\_world.sh", then we enter the following command:./hello\_world.sh and thanks to this our script works.

4.3. I know the following main components:

1)Ports: These allow you to connect peripherals such as a monitor, keyboard, and mouse;

2)Connectors: Connectors allow you to connect components such as the processor, memory, and drives.

3) Chipset: A chipset is a set of microcircuits that controls the interaction between the components of the motherboard

4) BIOS or UEFI: BIOS or UEFI is the software that is stored on the motherboard and is responsible for the initial boot of the computer.

5) Memory slots: The motherboard has memory slots where you can insert RAM modules. RAM is used to store data that the computer uses to run programs.

4.4. MBR is an older type of partition table supported by most operating systems. It can contain up to four main sections or up to three main sections and one extended section. An extended partition can be divided into logical partitions

GPT is a newer type of partition table supported by most modern operating systems. It has no limit on the number of partitions you can create.

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

* The essence of the mounting operation and why it is needed:

Access to other devices or data sources: One of the main purposes of mounting is the ability to interact with other devices or data sources. For example, you can mount external hard drives, USB drives, network resources, or CD/DVD drives to access their contents through the file system.

* Logical organization of data: Mounting allows you to logically organize data in a hierarchical structure. This means that you can display data from other sources as if they were in specific directories on your system.
* Data sharing: Mounting enables data sharing. For example, if you mount a network share, other users can also access that share through your system.

**Хід роботи**

**2.**

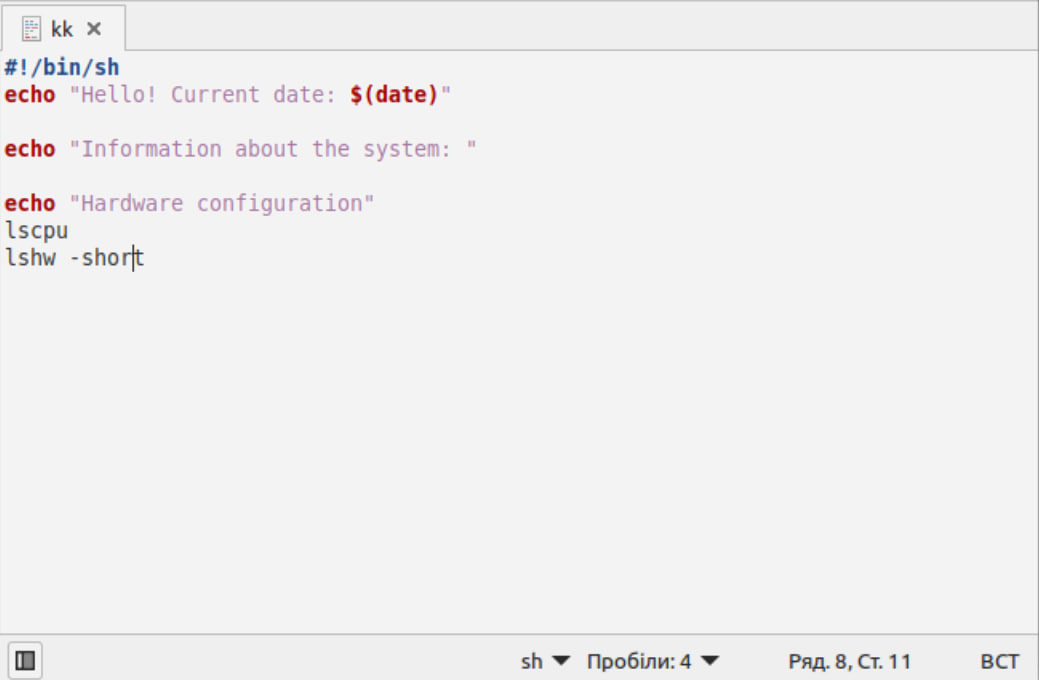
Lab 11: ( Когут )

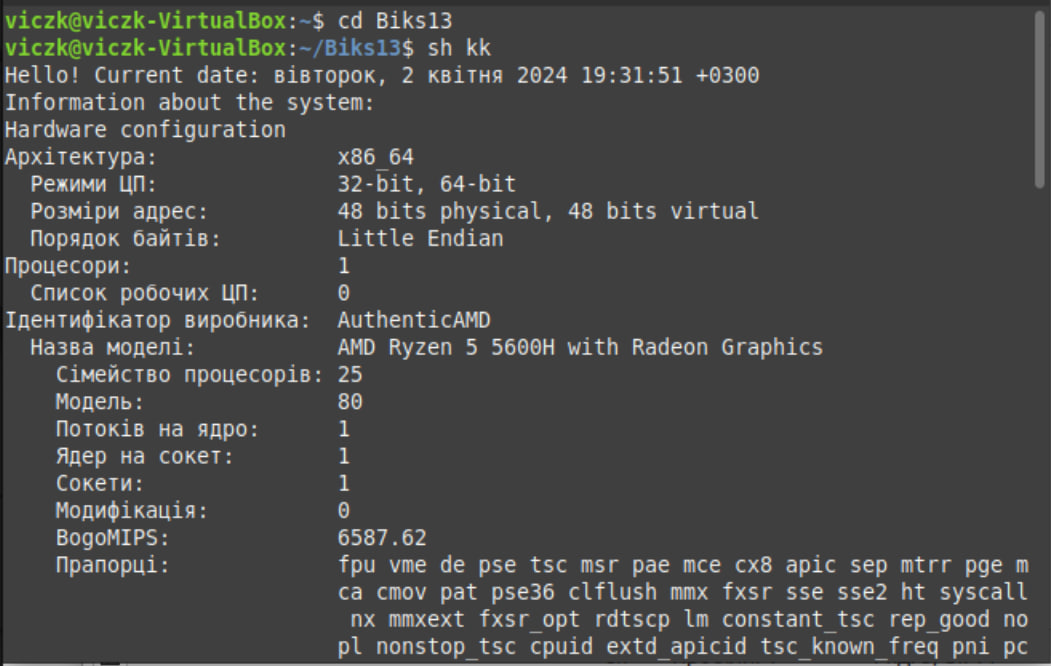
|  |  |
| --- | --- |
| Command name | Its purpose and functionality |
| vi | Text editor Vi, used for editing text files from the command line. |
| o | In the Vi editor, the o command adds a new line after the current line in insert mode. |
| O | In the Vi editor, the O command adds a new line before the current line in insert mode. |
| Esc | The Esc key exits insert mode and returns to command mode in the Vi editor. |
| :x, :wq | In the Vi editor, the :x or :wq command saves the file and exits the editor. |
| :q! | In the Vi editor, the :q! command exits the editor without saving changes. |
| :e! | In the Vi editor, the :e! command discards all changes and reloads the file. |
| :w! | In the Vi editor, the :w! command saves the file as "read-only" if possible. |
| cat | The cat command displays the contents of a file on the screen. |
| chmod | The chmod command is used to change permissions on files and directories. |
| echo | The echo command displays text or variable values on the screen. |
| test or [ ] | The test command or [ ] is used to perform conditional tests in scripts. |
| then | then marks the block of commands to be executed if the condition in the if statement is true. |
| else | else marks the block of commands to be executed if the condition in the if statement is false. |
| fi | fi marks the end of the block of commands in an if statement. |
| for | The for loop is used for iterating over a list of values and executing commands for each value. |
| do | do marks the block of commands executed in each iteration of a for loop. |
| done | done marks the end of the block of commands in a for loop. |
| seq | The seq command generates a sequence of numbers, helping with iteration in a for loop. |
| touch | The touch command creates empty files with specified names. |
| grep | The grep command is used to search for text in a file or standard input |
| /dev/null | /dev/null is a special file where output can be redirected when it is not needed. |

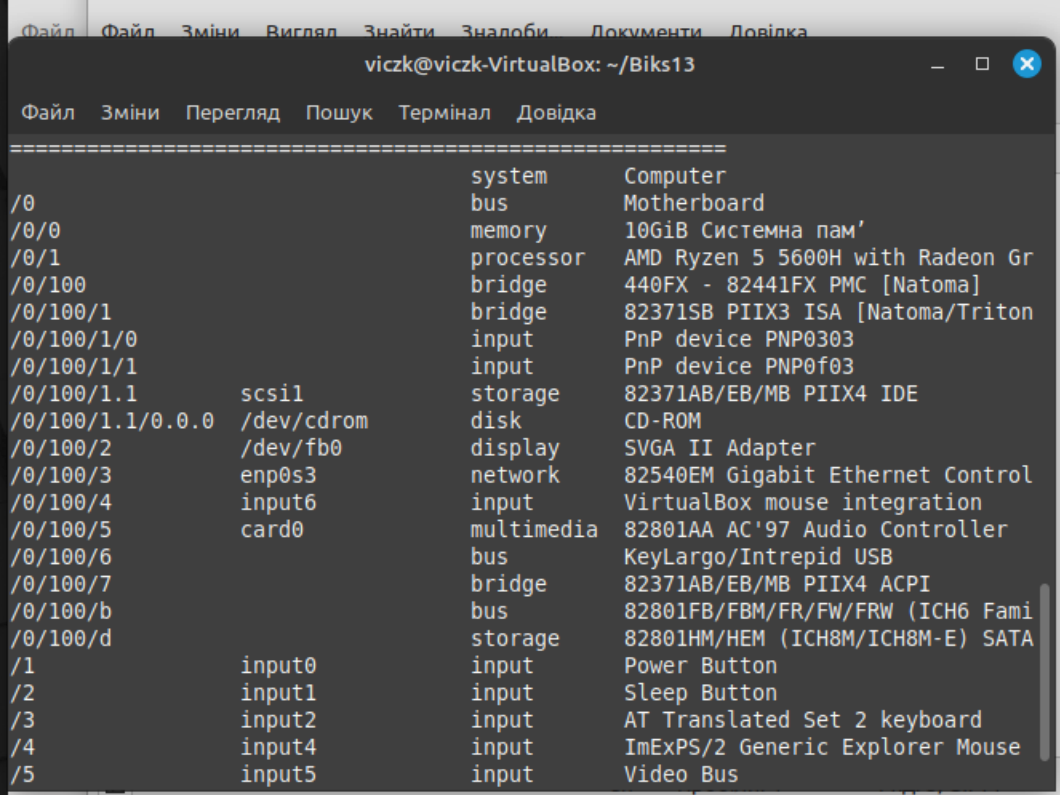
Lab 12: ( Береза )

|  |  |
| --- | --- |
| Command name | Its purpose and functionality |
| lscpu | Displays information about the CPU, including architecture, operating modes, number of cores, and more. |
| head -n 20 /proc/cpuinfo | Displays the first 20 lines of information from the /proc/cpuinfo file, including CPU details. |
| free -m | Shows information about RAM usage in megabytes, including total, used, free, and more. |
| free -g | Shows information about RAM usage in gigabytes, similar to free -m. |
| lspci | Lists devices connected to the PCI bus, including information about the manufacturer, model, and more. |
| lspci -k | Lists PCI devices along with the kernel drivers and modules used by them. |
| lsusb | Lists USB-connected devices, including device identifiers. |
| lsmod | Displays loaded kernel modules, their size, and dependencies on other modules. |
| fdisk -l | Shows information about disk partitions, including size, start and end sectors, and more. |

**3. ( Береза )**







**Контрольні запитання**

1. In Linux and the GNOME environment, you can work with variables and create branching and looping scenarios using shell scripting. You can declare and manipulate variables, use if statements for conditional scenarios, and employ for and while loops for repetitive tasks. Shell scripting uses different syntax and constructs compared to high-level programming languages, so consult the documentation of your specific shell (e.g., Bash, Zsh) for precise details.

2. In Linux, the arch command provides a basic overview of the system's CPU architecture, while the lscpu command offers detailed information about the CPU's characteristics and capabilities. arch gives a simple architecture description (e.g., x86\_64), while lscpu provides in-depth CPU-related details. Use the appropriate command based on your specific

information needs.

3. To obtain information about the current system's RAM usage in Linux (including the GNOME environment), you can use the free command. Here's a brief command:

free -h

This command will provide a summary of RAM usage in a human-readable format, showing details like total, used, free, and cached memory.

4. To view the status of connected peripheral devices in the Linux terminal (including the GNOME environment), you can use the following concise commands:

* lsusb: This command displays information about USB devices connected to your system.
* lspci: It shows information about PCI devices, such as graphics cards, network adapters, and other devices connected to your computer via the PCI bus.
* lsblk: This command provides information about block devices, such as hard drives and USB storage devices.
* lsscsi: For displaying information about SCSI devices, which can be used for external devices like optical drives.

These commands offer details about connected devices and their characteristics in your system.

5. GParted (GNOME Partition Editor) is a graphical partition management tool for Linux systems, specifically within the GNOME environment. Some of its key capabilities include:

* Partition Management: GParted allows you to create, delete, resize, and move partitions on your storage devices.
* File System Operations: It supports various file systems and enables actions like formatting partitions with different file systems.
* Partition Information: GParted provides detailed information about your partitions, including their size, file system, and usage.
* Partition Labeling: You can add labels to your partitions for easier identification.
* Partition Alignment: It helps in ensuring proper alignment of partitions for optimal performance.
* Data Recovery: GParted can be used for data recovery tasks, such as rescuing lost partitions or repairing damaged file systems.
* Bootloader Management: You can set or repair boot flags and manage bootloader configurations.
* Live CD/USB Support: GParted can be used from a live CD or USB to work on partitions without booting into the installed operating system.

Overall, GParted is a versatile tool for managing disk partitions and is particularly useful when you need to resize, create, or modify partitions on your Linux system.

**Сonclusion**

The objective of this work is to acquire practical skills in working with the Bash command shell and to familiarize oneself with the basic actions involved in creating script scenarios. To achieve these goals, the equipment used includes an IBM PC with a Windows operating system, a Virtual Box virtual machine running GNU/Linux CentOS, as well as online Linux courses on the Cisco Networking Academy website. Throughout the course of this work, students learn to execute commands in the Bash shell, create scripts for task automation, and gain practical skills in working with Linux.