“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 РПЗ-13б
Дімітрова С.П.
Перевірив викладач
Сушанова В.С.

Київ 2024

Робота студентки групи РПЗ-13б Дімітрової Софії

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.

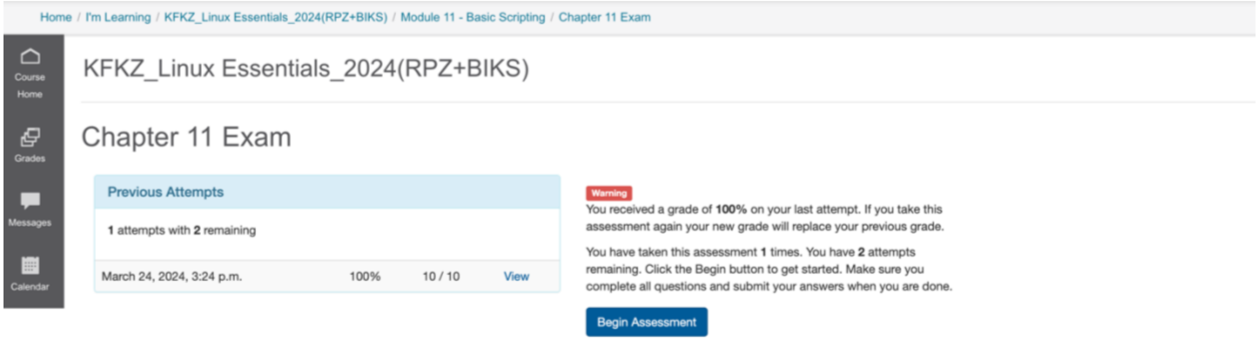
Термін англійською	Термін українською
Shell script	Скрипт оболонки
Script editor	Редактор сценаріїв
Repetitive	Повторюваний
Consistency	Послідовність
Execute	Виконати
Loop	Цикл
Shebang	Шебанг (#!)

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


Home / I'm Learning / KFKZ_Linux Essentials_2024(RPZ+BIKS) / Module 11 - Basic Scripting / Chapter 11 Exam
Course Home
Grades
Messages
Calendar
Warning
Begin Assessment

KFKZ_Linux Essentials_2024(RPZ+BIKS)

Chapter 11 Exam

Previous Attempts

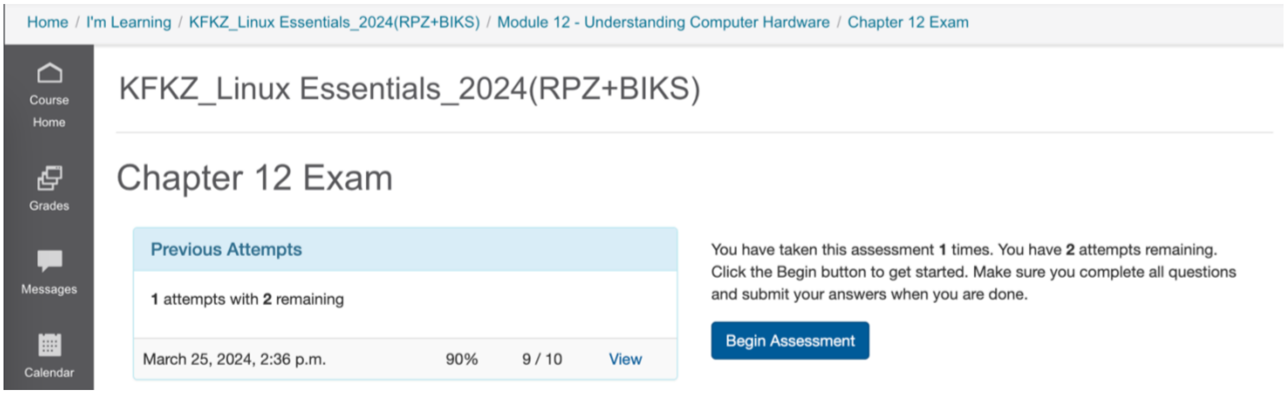
1 attempts with **2** remaining

March 24, 2024, 3:24 p.m.	100%	10 / 10	View

You received a grade of **100%** on your last attempt. If you take this assessment again your new grade will replace your previous grade.

You have taken this assessment **1** times. You have **2** attempts remaining. Click the Begin button to get started. Make sure you complete all questions and submit your answers when you are done.

- Chapter 12 Exam


Home / I'm Learning / KFKZ_Linux Essentials_2024(RPZ+BIKS) / Module 12 - Understanding Computer Hardware / Chapter 12 Exam
Course Home
Grades
Messages
Calendar
Begin Assessment

KFKZ_Linux Essentials_2024(RPZ+BIKS)

Chapter 12 Exam

Previous Attempts

1 attempts with **2** remaining

March 25, 2024, 2:36 p.m.	90%	9 / 10	View

You have taken this assessment **1** times. You have **2** attempts remaining. Click the Begin button to get started. Make sure you complete all questions and submit your answers when you are done.

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

4.1*Охарактеризуйте поняття скриптового сценарію у командній оболонці.

A shell script is a file that contains a set of commands that are executed one after the other when the script is run. Shell scripts have access to all shell commands, including logic. Thus, a script can check for a file or look for a certain output and change its behaviour accordingly. Scripts automate repetitive tasks, improve consistency, and can perform complex operations.

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

Scripts are created using a text editor like nano or vim. Save the script with a .sh extension. For example, to create a new script using the Nano editor, simply type *nano myscript.sh*. This will open the Nano text editor and create a new file called "myscript.sh". Next, enter the required code for your script in this file.
There are two ways to run a script:

- Directly from the shell: ./myscript.sh. This requires the script to have execute permissions. You can set them using chmod +x myscript.sh.
- As an argument to the shell: sh myscript.sh (replace sh with your shell if different, e.g., bash script.sh).

4.3**Які основні компоненти материнської плати ви знаєте?

The main components of a motherboard are the central processing unit (CPU) socket, memory slots (RAM), expansion slots, chipset, input/output (I/O) ports, power connectors, and battery.

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

The concepts of MBR (Master Boot Record) and GPT (GUID Partition Table) are used for partitioning and managing disk space on storage devices such as hard disk drives (HDDs), solid-state drives (SSDs), USB flash drives, and other storage media. They determine how the disk is organized into partitions, allowing users to store data, install operating systems, and manage storage efficiently.

MBR is used for older systems. It has a limit of 4 partitions and does not support drives larger than 2 TB. GPT is a modern format that allows for more partitions, supports larger drives, and provides reliability.

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

The essence of the mounting operation in operating systems like Linux is to integrate a filesystem, stored on a storage device (such as a hard drive or USB flash drive), into the directory tree of the operating system. Once mounted, the filesystem becomes accessible through a designated path within the system. Mounting is necessary to access the data stored on the storage device. It allows the operating system to interact with the files and directories within the filesystem, enabling users to read, write, and manipulate data. Without mounting, the data on the storage device would remain inaccessible to the operating system and its users.

Хід роботи:

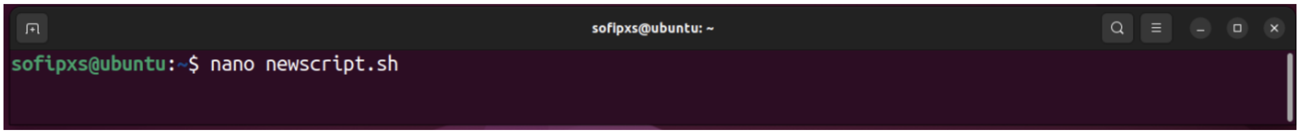
1. Початкова робота в CLI-режимі в Linux ОС сімейства Linux:
   1.1. Запустіть віртуальну машину VirtualBox, оберіть CentOS та запустіть її. Виконайте вхід в систему під користувачем: CentOS, пароль для входу: reverse ***(якщо виконуєте ЛР у 401 ауд.)*** та запустіть термінал.
   1.2. Запустіть віртуальну 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
Lab 11 Basic Scripting	
vi myfile	Create a new file.
wq	Write the file to disk and quit.
j	Moves cursor down one line (same as down arrow)
k	Moves cursor up line (same as up arrow)
l	Moves cursor to the right one character (same as right arrow)
h	Moves cursor to the left one character (same as left arrow).
w	Moves cursor to beginning of next word.
e	Moves cursor to end of word.
b	Moves cursor to beginning of previous word.
$	Moves cursor to end of current line (same as End key).
0 (zero)	Moves cursor beginning of current line (same as Home key).
3G	Jumps to third line (nG jumps to the nth line).
1G	Jumps to first line.
Shift + G	Jumps to the last line.
dw	Delete the word
u	Undo the last operation.
2dw	Delete two words.
xxxx	Delete four characters, one at a time.
4u	Undo the last 4 operations and recover the deleted characters.
14x	Delete 14 characters.
5x	Delete the five characters to the left of the cursor (type 5 then Shift+x).
dd	Delete the current line.
p	Whatever was lasted deleted or yanked can be “pasted”. Paste the deleted lines below the current line.
2u	Undo the last two operations
2dd	Delete two lines, the current and the next.
4w D	Move to the fourth word then delete from the current position to the end of the line Shift+D.
J	Join two lines, the current and the next by typing a capital J (Shift+J).
yw	Copy (or “yank”) the current word.
P	Paste (or “put”) the copied word before the current cursor by typing Shift+p.
1G 3J	Move to the first line, then join three lines.

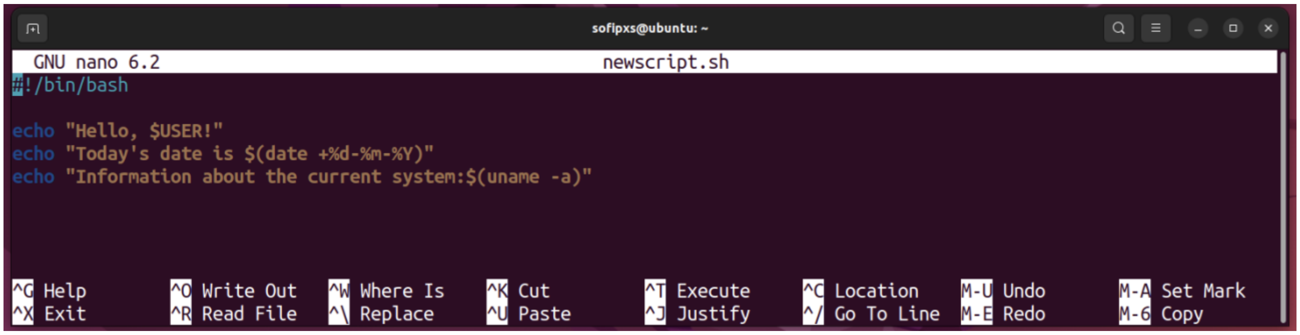
:%s/text //g	Search for and delete the word text (add a space after the word text).
i	Enter insert mode
~	Shift+` changes letter to lower case
:w	Save the file.
10l	10 followed by the lowercase ‘L’
a	Enter insert mode.
o	Open a blank line below the current line.
O	Open a blank line above the current line by pressing uppercase O.
:x	Will save and close the file.
:wq	Will write to file and quit.
:wq!	Will write to a read-only file, if possible, and quit.
ZZ	Will save and close. Notice that no colon : is used in this case.
:q!	Exit without saving changes.
:e!	Discard changes and reload file.
:w!	Write to read-only, if possible.
/line	Search forward for the word line. The cursor moves to the beginning of the first instance of the word line.
n	Search for the next instance of the word line.
?line	Search backward for the word line. Cursor moves to the beginning of the previous instance of the word line.
cw entry	You will replace the word line with the word entry. When you press cw you will be in insert mode and you will be able to type over the word line.
bash sample.sh	Run bash script.
chmod a+x	Give some new rules to script.
Lab12 Understanding Computer Hardware	
lscpu	In order to determine the type of CPU execute the lscpu command.
head -n 20 /proc/cpuinfo	Use the head command with the -n option to list the first 20 lines of the cpuinfo file.
head -n 20 /proc/cpuinfo	Use the head command with the -n option to list the first 20 lines of the cpuinfo file.
free -m free -g	To discover how much RAM and swap space is being used, use the free command. The output shows the amount of memory in megabytes when the -m option is used and in gigabytes when the -g option is used:
lspci	To see what devices are connected to the PCI bus, use the lspci

	command.
lspci -k	lspci command with the -k option shows devices along with the kernel driver and modules used
lsusb	Attempt to list the USB connected devices.
lsmod	For hardware to function, the Linux kernel usually loads a driver or module. Use the lsmod command to view the currently loaded modules.
fdisk	The fdisk command is useful for identifying and manipulating disk storage resources on a system. Without the -l option, the fdisk command enters an interactive mode that is typically used to modify partitions on a disk device.
fdisk -l	The -l option lists the partition tables for the specified devices and then exits.

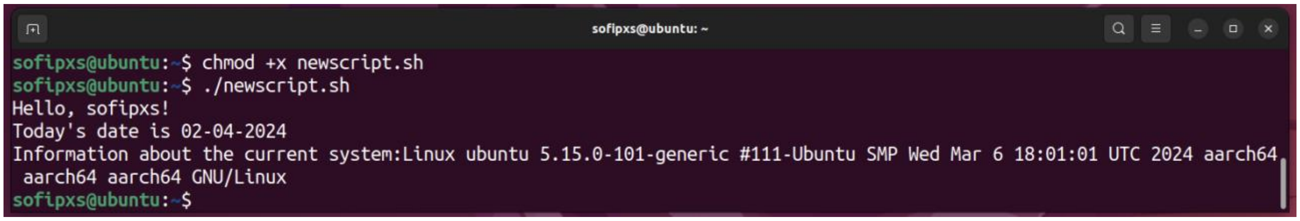
3. Створіть скриптові сценарії з виводом текстових повідомлень для користувача (продемонструйте скріншоти):
   - сценарій має виводити привітання до поточного користувача вказуючи поточну дату та інформацію про поточну систему;



```
sofipxs@ubuntu:~$ nano newscript.sh
```



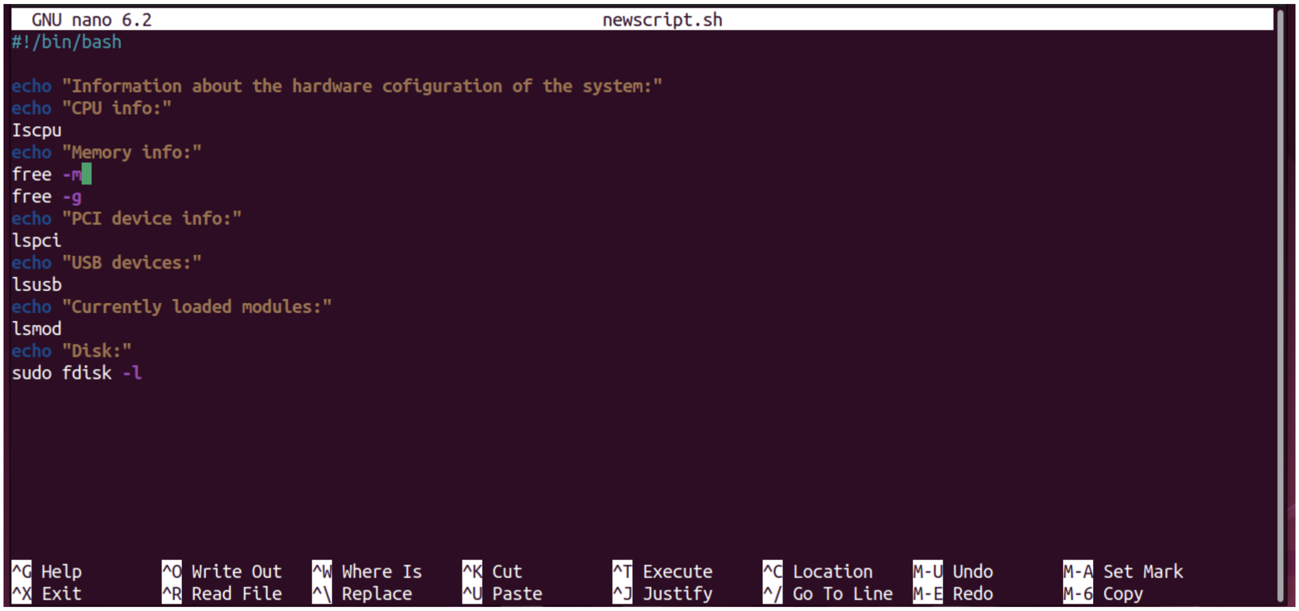
```
  GNU nano 6.2                              newscript.sh
#!/bin/bash

echo "Hello, $USER!"
echo "Today's date is $(date +%d-%m-%Y)"
echo "Information about the current system:$(uname -a)"
```



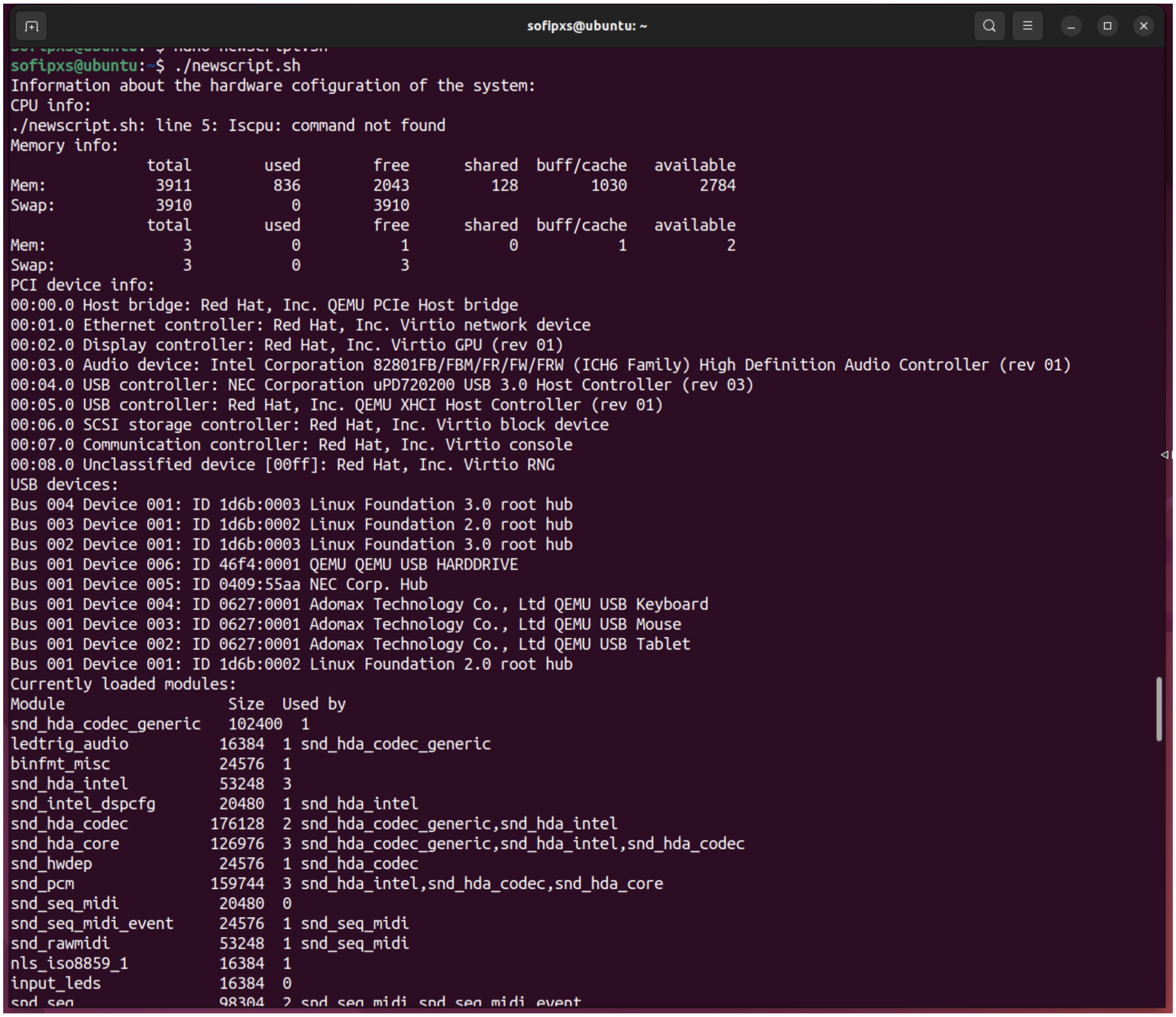
```
sofipxs@ubuntu:~$ chmod +x newscript.sh
sofipxs@ubuntu:~$ ./newscript.sh
Hello, sofipxs!
Today's date is 02-04-2024
Information about the current system:Linux ubuntu 5.15.0-101-generic #111-Ubuntu SMP Wed Mar 6 18:01:01 UTC 2024 aarch64
 aarch64 aarch64 GNU/Linux
sofipxs@ubuntu:~$
```

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

Робота студентки групи РПЗ-13б Дімітрової Софії

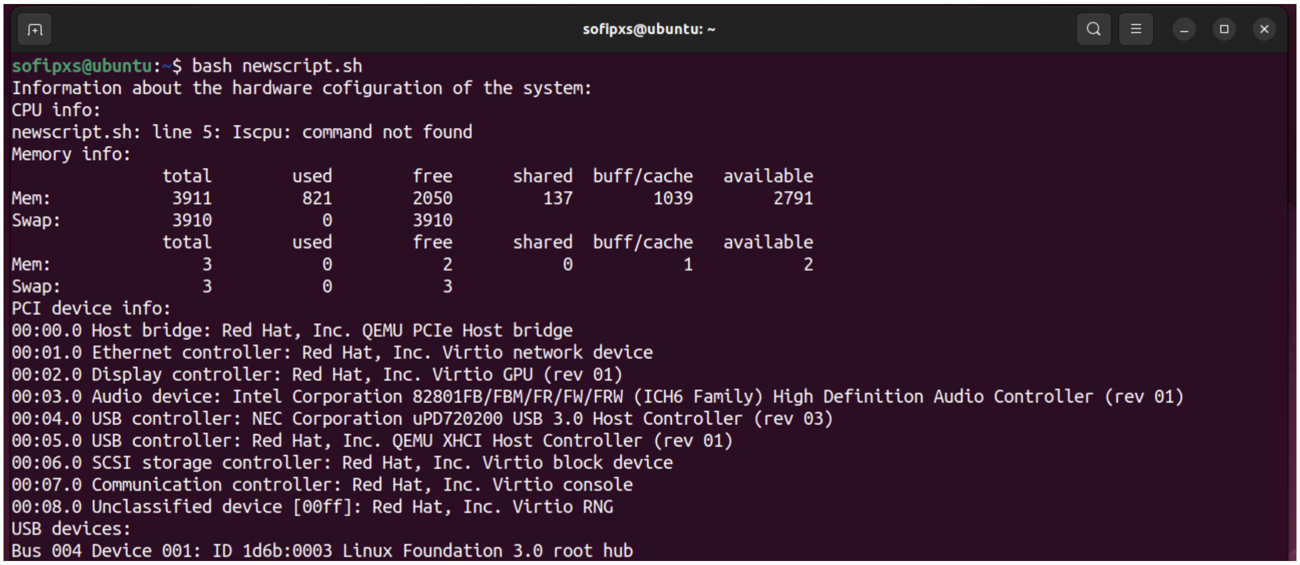


```
  GNU nano 6.2                               newscript.sh
#!/bin/bash

echo "Information about the hardware cofiguration of the system:"
echo "CPU info:"
Iscpu
echo "Memory info:"
free -m
free -g
echo "PCI device info:"
lspci
echo "USB devices:"
lsusb
echo "Currently loaded modules:"
lsmod
echo "Disk:"
sudo fdisk -l
```



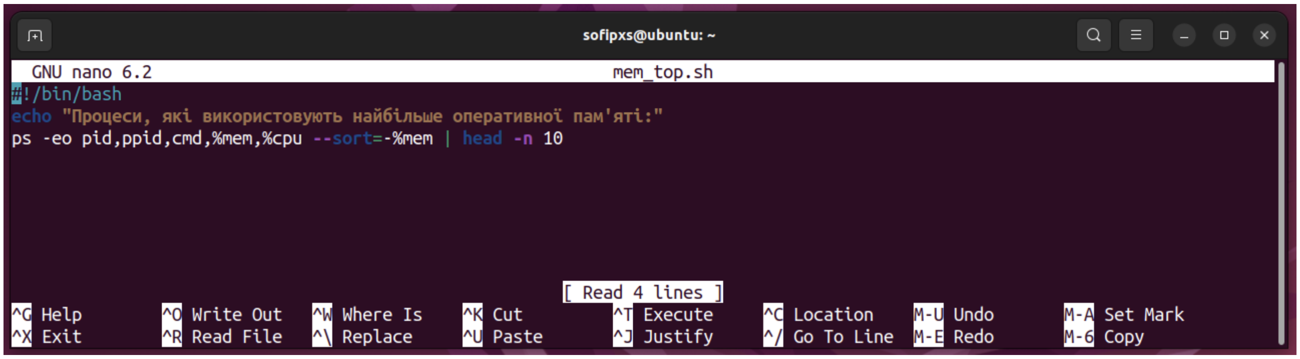
```
sofipxs@ubuntu:~$ ./newscript.sh
Information about the hardware cofiguration of the system:
CPU info:
./newscript.sh: line 5: Iscpu: command not found
Memory info:
               total        used        free      shared  buff/cache   available
Mem:            3911         836        2043         128        1030        2784
Swap:           3910           0        3910
               total        used        free      shared  buff/cache   available
Mem:               3           0           1           0           1           2
Swap:              3           0           3
PCI device info:
00:00.0 Host bridge: Red Hat, Inc. QEMU PCIe Host bridge
00:01.0 Ethernet controller: Red Hat, Inc. Virtio network device
00:02.0 Display controller: Red Hat, Inc. Virtio GPU (rev 01)
00:03.0 Audio device: Intel Corporation 82801FB/FBM/FR/FW/FRW (ICH6 Family) High Definition Audio Controller (rev 01)
00:04.0 USB controller: NEC Corporation uPD720200 USB 3.0 Host Controller (rev 03)
00:05.0 USB controller: Red Hat, Inc. QEMU XHCI Host Controller (rev 01)
00:06.0 SCSI storage controller: Red Hat, Inc. Virtio block device
00:07.0 Communication controller: Red Hat, Inc. Virtio console
00:08.0 Unclassified device [00ff]: Red Hat, Inc. Virtio RNG
USB devices:
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 006: ID 46f4:0001 QEMU QEMU USB HARDDRIVE
Bus 001 Device 005: ID 0409:55aa NEC Corp. Hub
Bus 001 Device 004: ID 0627:0001 Adomax Technology Co., Ltd QEMU USB Keyboard
Bus 001 Device 003: ID 0627:0001 Adomax Technology Co., Ltd QEMU USB Mouse
Bus 001 Device 002: ID 0627:0001 Adomax Technology Co., Ltd QEMU USB Tablet
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Currently loaded modules:
Module                  Size  Used by
snd_hda_codec_generic   102400  1
ledtrig_audio          16384  1 snd_hda_codec_generic
binfmt_misc            24576  1
snd_hda_intel          53248  3
snd_intel_dspcfg       20480  1 snd_hda_intel
snd_hda_codec         176128  2 snd_hda_codec_generic,snd_hda_intel
snd_hda_core          126976  3 snd_hda_codec_generic,snd_hda_intel,snd_hda_codec
snd_hwdep              24576  1 snd_hda_codec
snd_pcm               159744  3 snd_hda_intel,snd_hda_codec,snd_hda_core
snd_seq_midi           20480  0
snd_seq_midi_event     24576  1 snd_seq_midi
snd_rawmidi            53248  1 snd_seq_midi
nls_iso8859_1          16384  1
input_leds             16384  0
snd_seq                98304  2 snd_seq_midi,snd_seq_midi_event
```

Another way to run the program is to type bash before the filename:

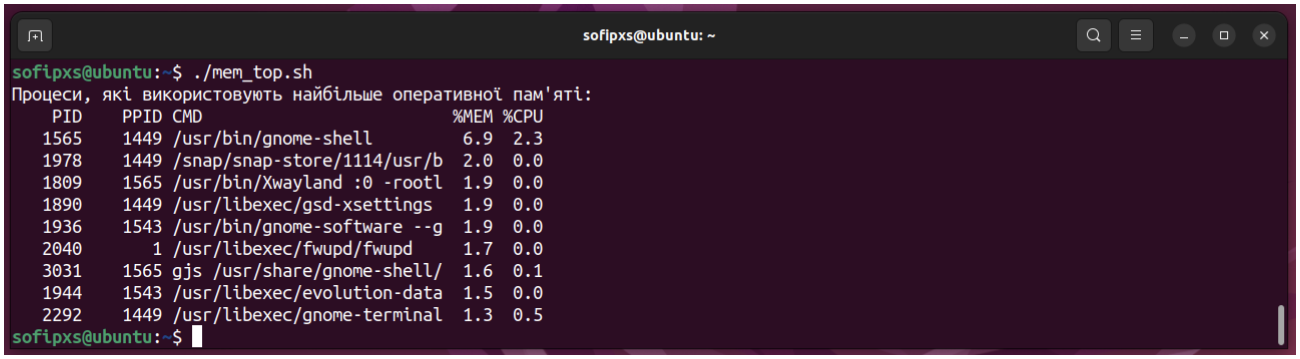


```
sofipxs@ubuntu:~$ bash newscript.sh
Information about the hardware cofiguration of the system:
CPU info:
newscript.sh: line 5: Iscpu: command not found
Memory info:
               total        used        free      shared  buff/cache   available
Mem:            3911         821        2050         137        1039        2791
Swap:           3910           0        3910
               total        used        free      shared  buff/cache   available
Mem:               3           0           2           0           1           2
Swap:              3           0           3
PCI device info:
00:00.0 Host bridge: Red Hat, Inc. QEMU PCIe Host bridge
00:01.0 Ethernet controller: Red Hat, Inc. Virtio network device
00:02.0 Display controller: Red Hat, Inc. Virtio GPU (rev 01)
00:03.0 Audio device: Intel Corporation 82801FB/FBM/FR/FW/FRW (ICH6 Family) High Definition Audio Controller (rev 01)
00:04.0 USB controller: NEC Corporation uPD720200 USB 3.0 Host Controller (rev 03)
00:05.0 USB controller: Red Hat, Inc. QEMU XHCI Host Controller (rev 01)
00:06.0 SCSI storage controller: Red Hat, Inc. Virtio block device
00:07.0 Communication controller: Red Hat, Inc. Virtio console
00:08.0 Unclassified device [00ff]: Red Hat, Inc. Virtio RNG
USB devices:
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
```

- **наведіть свій приклад скриптового сценарію.

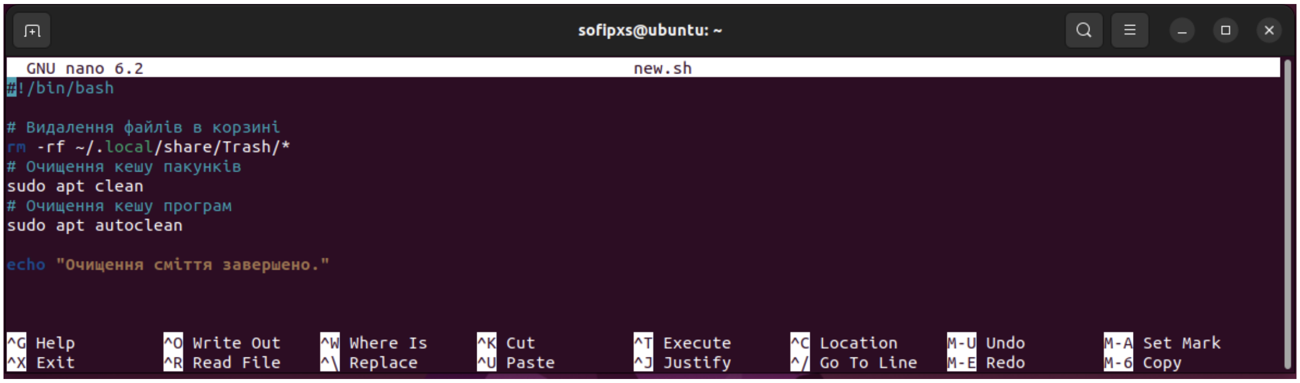


```
#!/bin/bash
echo "Процеси, які використовують найбільше оперативної пам'яті:"
ps -eo pid,ppid,cmd,%mem,%cpu --sort=-%mem | head -n 10
```

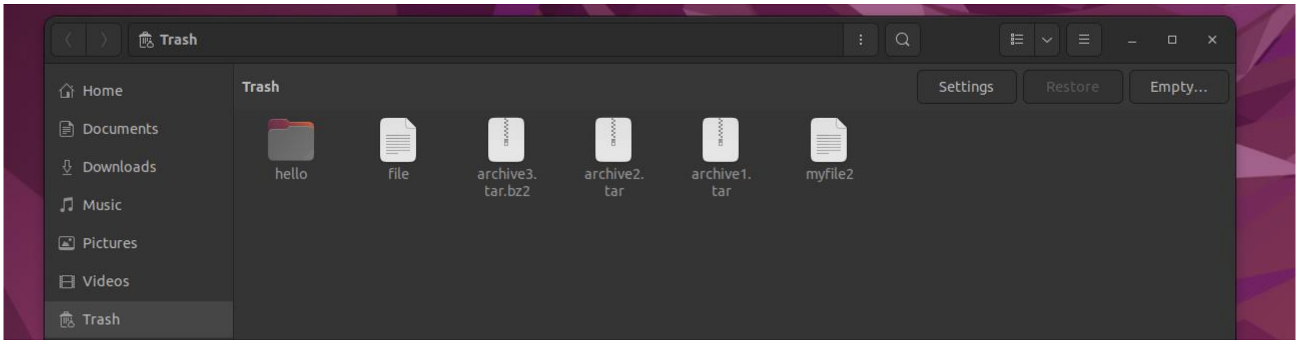


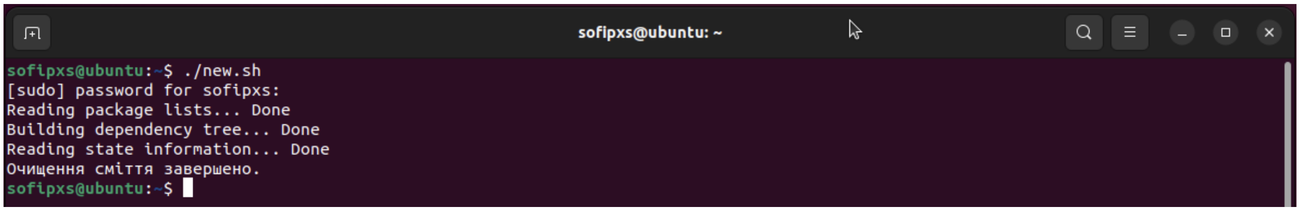
```
sofipxs@ubuntu:~$ ./mem_top.sh
Процеси, які використовують найбільше оперативної пам'яті:
    PID    PPID CMD                         %MEM %CPU
   1565    1449 /usr/bin/gnome-shell         6.9  2.3
   1978    1449 /snap/snap-store/1114/usr/b  2.0  0.0
   1809    1565 /usr/bin/Xwayland :0 -rootl  1.9  0.0
   1890    1449 /usr/libexec/gsd-xsettings   1.9  0.0
   1936    1543 /usr/bin/gnome-software --g  1.9  0.0
   2040       1 /usr/libexec/fwupd/fwupd     1.7  0.0
   3031    1565 gjs /usr/share/gnome-shell/  1.6  0.1
   1944    1543 /usr/libexec/evolution-data  1.5  0.0
   2292    1449 /usr/libexec/gnome-terminal  1.3  0.5
sofipxs@ubuntu:~$ 
```

The second script, which deletes files from the Recycle Bin, clears the package cache and also clears the application cache.

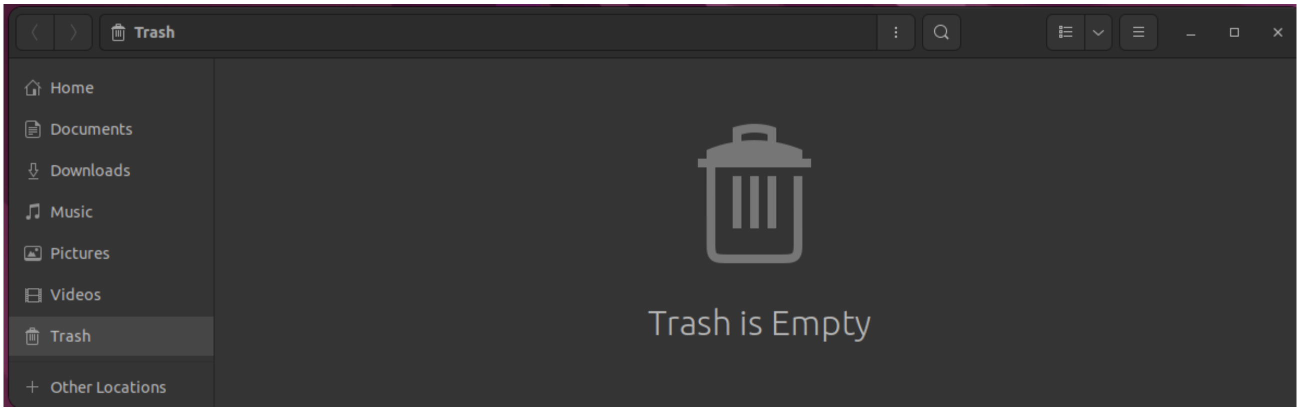


```
#!/bin/bash

# Видалення файлів в корзині
rm -rf ~/.local/share/Trash/*
# Очищення кешу пакунків
sudo apt clean
# Очищення кешу програм
sudo apt autoclean

echo "Очищення сміття завершено."
```





```
sofipxs@ubuntu:~$ ./new.sh
[sudo] password for sofipxs:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Очищення сміття завершено.
sofipxs@ubuntu:~$
```

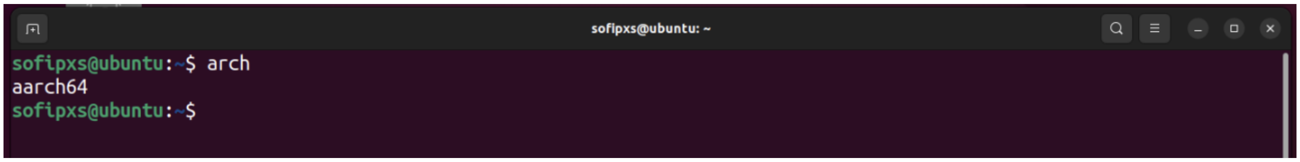


Trash is Empty

Відповіді на контрольні запитання:

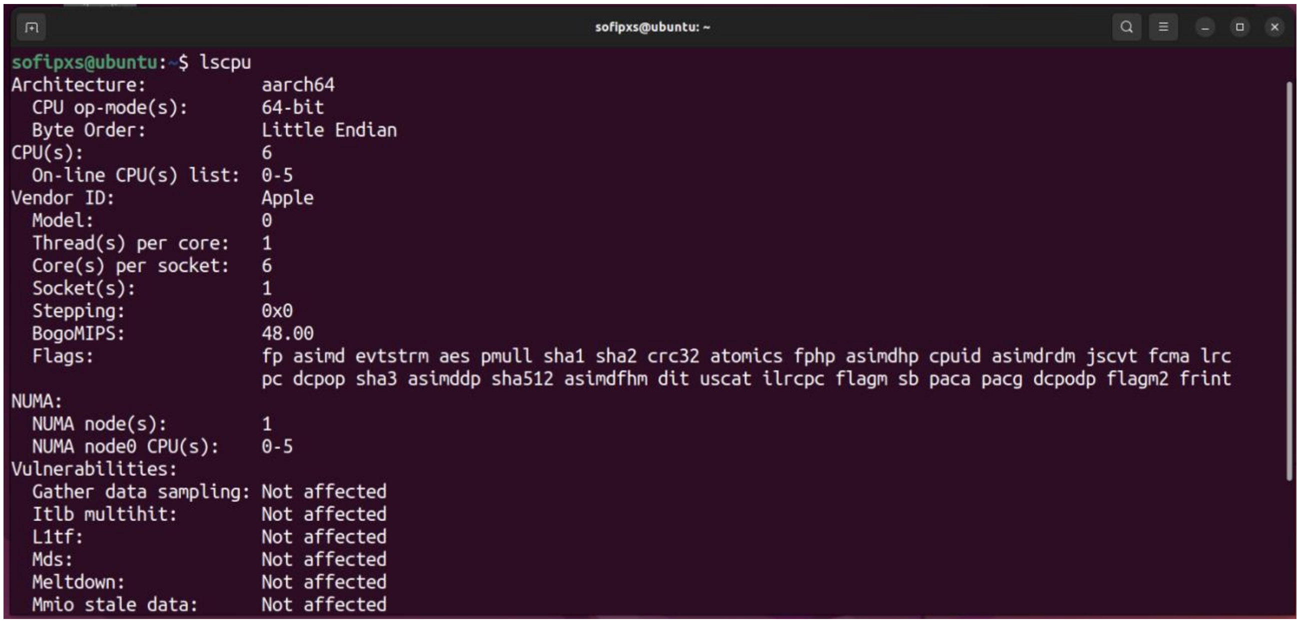
1. В чому відмінність між командами arch та lscpu?

    - The arch command in Linux is used to print the computer's architecture. It's a simple command with no options or arguments.



```
sofipxs@ubuntu:~$ arch
aarch64
sofipxs@ubuntu:~$
```

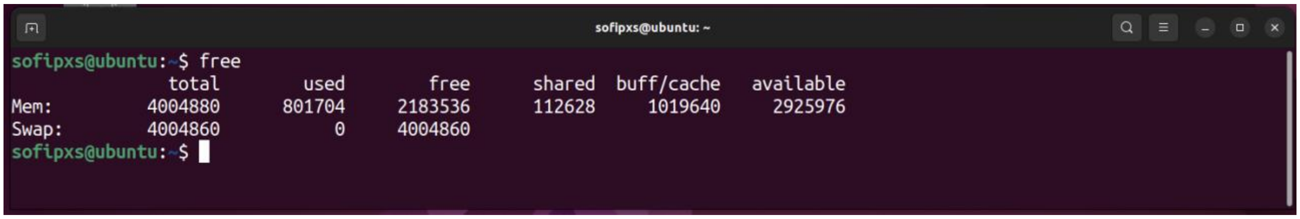
- The lscpu command, on the other hand, provides a detailed list of information about the CPUs in the system. In this case, the output details the CPU architecture, cores, threads, CPU family, model, etc.



```
sofipxs@ubuntu:~$ lscpu
Architecture:           aarch64
  CPU op-mode(s):       64-bit
  Byte Order:           Little Endian
CPU(s):                 6
  On-line CPU(s) list:  0-5
Vendor ID:              Apple
  Model:                0
  Thread(s) per core:   1
  Core(s) per socket:   6
  Socket(s):            1
  Stepping:             0x0
  BogoMIPS:             48.00
  Flags:                fp asimd evtstrm aes pmull sha1 sha2 crc32 atomics fphp asimdhp cpuid asimdrdm jscvt fcma lrc
                        pc dcpop sha3 asimddp sha512 asimdfhm dit uscat ilrcpc flagm sb paca pacg dcpodp flagm2 frint
NUMA:
  NUMA node(s):         1
  NUMA node0 CPU(s):    0-5
Vulnerabilities:
  Gather data sampling: Not affected
  Itlb multihit:        Not affected
  L1tf:                 Not affected
  Mds:                  Not affected
  Meltdown:             Not affected
  Mmio stale data:      Not affected
```

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

   The free command provides information on total, used, free, shared, buffer/cache, and available RAM and swap space at the moment you run the command:



```
sofipxs@ubuntu:~$ free
               total        used        free      shared  buff/cache   available
Mem:         4004880      801704     2183536      112628     1019640     2925976
Swap:        4004860           0     4004860
sofipxs@ubuntu:~$ 
```

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

In command-line scripts such as Bash, variables are manipulated by assigning values and using them in commands and expressions. To create branched scripts, conditional statements such as *if*, *elif*, and *else* are used to perform different actions depending on the conditions. Loop scripts are created using loops, such as *for*, *while*, which allow you to execute a certain block of code several times. Such constructs allow you to automate and control the execution of scripts, ensuring their flexibility and functionality.

Simple use of variables:

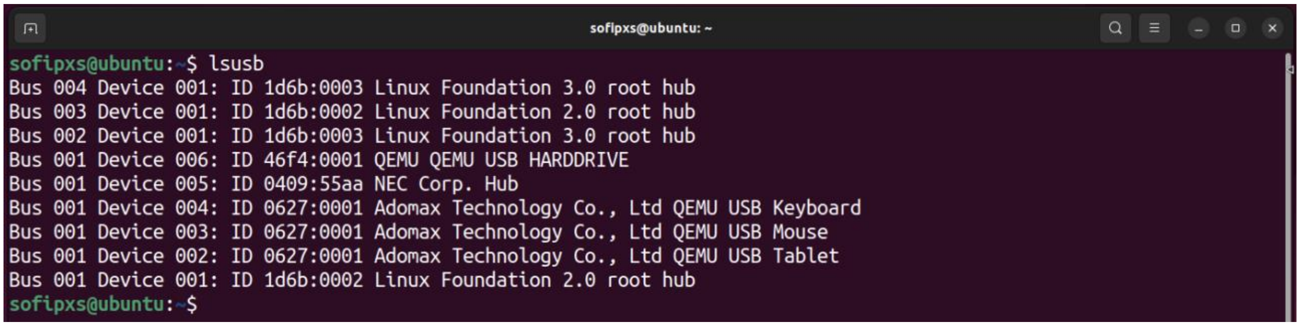
```
#!/bin/bash

ANIMAL="penguin"
echo "My favorite animal is a $ANIMAL"
```

A basic if statement looks like this:

```
if somecommand; then
  # do this if somecommand has an exit code of 0
fi
```

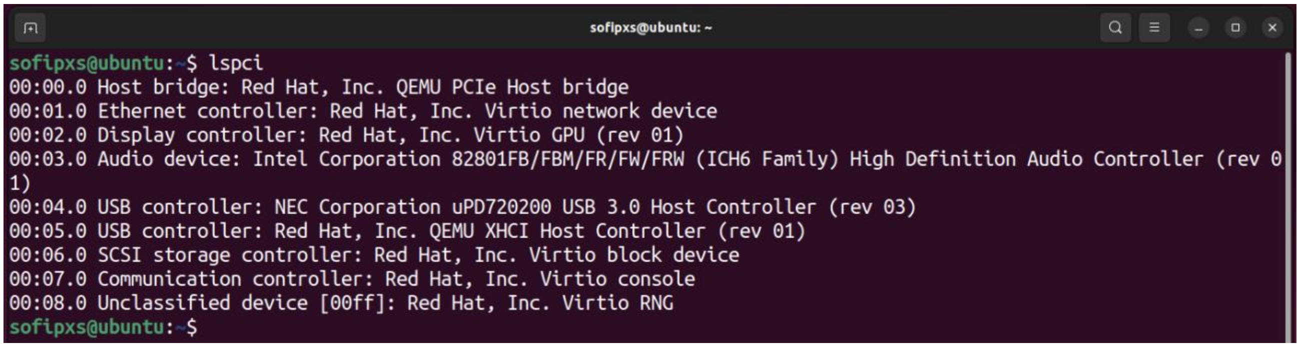
4. *Які команди для перегляду стану підключення периферійних пристроїв можна використати в терміналі?

- lsusb: This command displays a list of all USB devices connected to the computer along with their IDs.



```
sofipxs@ubuntu:~$ lsusb
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 006: ID 46f4:0001 QEMU QEMU USB HARDDRIVE
Bus 001 Device 005: ID 0409:55aa NEC Corp. Hub
Bus 001 Device 004: ID 0627:0001 Adomax Technology Co., Ltd QEMU USB Keyboard
Bus 001 Device 003: ID 0627:0001 Adomax Technology Co., Ltd QEMU USB Mouse
Bus 001 Device 002: ID 0627:0001 Adomax Technology Co., Ltd QEMU USB Tablet
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
sofipxs@ubuntu:~$
```

- lspci: This command displays information about all PCI devices that are connected to the system.



```
sofipxs@ubuntu:~$ lspci
00:00.0 Host bridge: Red Hat, Inc. QEMU PCIe Host bridge
00:01.0 Ethernet controller: Red Hat, Inc. Virtio network device
00:02.0 Display controller: Red Hat, Inc. Virtio GPU (rev 01)
00:03.0 Audio device: Intel Corporation 82801FB/FBM/FR/FW/FRW (ICH6 Family) High Definition Audio Controller (rev 0
1)
00:04.0 USB controller: NEC Corporation uPD720200 USB 3.0 Host Controller (rev 03)
00:05.0 USB controller: Red Hat, Inc. QEMU XHCI Host Controller (rev 01)
00:06.0 SCSI storage controller: Red Hat, Inc. Virtio block device
00:07.0 Communication controller: Red Hat, Inc. Virtio console
00:08.0 Unclassified device [00ff]: Red Hat, Inc. Virtio RNG
sofipxs@ubuntu:~$
```

- lsblk: Displays a list of all block devices, such as hard drives and USB drives, along with information about their size and mount points.

- lshw: This command provides detailed information about all hardware components of the system, including peripherals.

5. **Які можливості застунку gparted?

The gparted program is a GNOME partition editor designed to create, rearrange, and remove partitions on a disc. A disc device can be partitioned into one or more partitions. The gparted application allows you to reorder the partitions on a disc device, while preserving the data stored on the partitions. It supports many file system types including ext2, ext3, ext4, fat16, fat32, ntfs, and others. With gparted, you can perform the following tasks:

- Create a partition table on a disc device.
- Enable or disable partition flags, including the boot and hide flags.
- Perform actions on partitions, including creating, deleting, resizing, moving, verifying, labelling, copying, and pasting partitions.

Висновки:

During the laboratory work, I got acquainted with the basic concepts of working with the Bash command shell and writing scripts in the GNU/Linux environment. I gained practical skills in creating and editing scripts. The main components of the motherboard, the essence of the mount operation, and ways to obtain information about the system's hardware configuration were discussed.