“Київський фаховий коледж зв’язку”

Циклова комісія Комп’ютерної інженерії

**ЗВІТ ПО ВИКОНАННЮ**

**ЛАБОРАТОРНОЇ РОБОТИ №6**

з дисципліни: «Операційні системи»

**Тема: «Команди Linux для архівування та стиснення даних. Робота з текстом»**

Виконавли студенти

групи РПЗ-03

Команда: Губенко Є.О.,

Заїка С.В. та Кресан Р.А.

Перевірив викладач

Сушанова В.С.

Київ 2022

**Мета роботи:**

1. Отримання практичних навиків роботи з командною оболонкою Bash.
2. Знайомство з базовими командами для архівування та стиснення даних.
3. Знайомство з базовими діями при роботі з текстом у терміналі.

**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**

1. Прочитайте короткі теоретичні відомості до лабораторної роботи та зробіть невеликий словник базових англійських термінів з питань призначення команд та їх параметрів.
2. Вивчіть матеріали онлайн-курсу академії Cisco “NDG Linux Essentials”:

* Chapter 09 - Archiving and Compression
* Chapter 10 - Working With Text

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

* Chapter 09 Exam
* Midterm Exam (Modules 1 - 9) буде окреме завдання в гугл-класі
* Chapter 10 Exam

***Готував матеріал студент Заїка С.В.***

1. На базі розглянутого матеріалу дайте відповіді на наступні питання:
   1. Яке призначення команд tar, xz, zip, bzip, gzip? Зробіть короткий опис кожної команди та виділіть їх основні параметри. Яким чином їх можна встановити.

*1) The "tar" command is used to create and unpack archives in tar format. The main parameters:*

*c: create a new archive*

*x: unpack the archive*

*v: display the list of files during the operation*

*f: specify the name of the file to which the archive will be written*

*To install tar on most Linux systems, run the command:*



1. The "xz" command is used to compress files in the xz format. Main parameters:

z: compress the file

d: unzip the file

To install xz on most Linux systems, run the command:



1. 2) The "zip" command is used to create and unzip archives in zip format. Main parameters:

r: add files to the archive recursively

e: compress files with encryption

d: remove files from the archive

To install zip on most Linux systems, run the command:



4) The "bzip" command is used to compress files in bzip format. Main parameters:

z: compress the file

d: unzip the file

To install bzip on most Linux systems, run the command:



5) The "gzip" command is used to compress files in the gzip format. Main parameters:

z: compress the file

d: unzip the file

To install gzip on most Linux systems, run the command:



* 1. Наведіть три приклади реалізації архівування та стискання даних різними командами.

1. *Create a tar archive and compress it with gzip:*



*This command creates an archive named "archive.tar.gz" that contains files and directories located at "/path/to/directory" and compresses it using gzip.*

1. *Compressing a file with bzip:*



*This command compresses a file named "file.txt" using bzip2 and saves it as "file.txt.bz2".*

1. *Create a zip archive:*



*This command creates a zip archive named "archive.zip" that contains files and directories located at "/path/to/directory ".*

* 1. Яке призначення команд cat, less, more, head and tail? Зробіть короткий опис кожної команди та виділіть їх основні параметри. Яким чином їх можна встановити?

*cat: this command is used to display the contents of files on the screen.*

*The basic syntax of the cat command is:*



Main parameters:

-n: prints line numbers;

-b: prints numbers only for non-empty lines.

*1) less: this command is designed to display the contents of files gradually with the ability to scroll in both directions.*

*The basic syntax of the less command is:*



Main parameters*:*

*q: exit the program;*

*f: scroll forward one screen;*

*b: scroll back one screen;*

*/pattern: search for the specified pattern in the file.*

1. *more: this command is also designed to display the contents of files gradually with the ability to scroll in both directions, but is less functional than less.*

*The basic syntax of the command more:*

**

Main parameters*:*

*q: вихід з програми;*

*f: прокручування вперед на один екран;*

*b: прокручування назад на один екран.*

1. *head: this command is designed to display the first few lines of a file.*

*The basic syntax of the command head:*

**

Main parameters*:*

*-n [number]: displays the first n lines of the file.*

1. *tail: this command is designed to display the last few lines of a file.*

*The basic syntax of the command tail:*

**

Main parameters*:*

*-* *n [number]: prints the last n lines of the file.*

* 1. Поясніть принципи роботи командної оболонки з каналами, потоками та фільтрами.

*The Linux command shell has a number of options for working with channels, streams, and filters, which greatly facilitates the work with data and ensures its optimal processing.*

*Pipelines allow you to pass the output of one command as input to another command without having to save the result of the first command to a file. The syntax for pipes consists of a vertical slash |, which indicates that the output of the left command is passed as input to the right command. For example, the ls | grep file command will return a list of files in the current directory containing the word "file".*

*Streams in Linux can be entered from the keyboard (stdin), displayed on the screen (stdout), or sent to a file (stderr). Special characters are used to work with streams. For example, the > character is used to redirect the output of a command to a file, the >> character to append an existing file, the < character to use a file as input to a command.*

*Filters are commands that read input data from the standard input (stdin), process it, and output the result to the standard output (stdout). Filters can be combined with channels and redirects to get the desired result. Some of the most commonly used filters in Linux include grep to search for lines in a file, sed to edit and replace text, awk to process and format text.*

*In general, the command shell in Linux allows you to easily and efficiently work with data streams using channels and filters, which helps to facilitate work with files and data.*

* 1. Яке призначення команди grep?

*The grep command is one of the most powerful and popular commands in Unix-like operating systems, used to search input data for a specific text pattern. The grep command is often used to search for large amounts of text, such as log files and other texts stored as data streams.*

*The basic syntactic structure of the grep command is as follows:*



*The main parameters of the grep command:*

*-i - ignore case;*

*-v - return the lines that do not match the pattern;*

*-n - return the line numbers of the lines containing the match;*

*-r - recursive search in subdirectories.*

*The template can be plain text or a regular expression to more precisely define the search text.*

*For example, the grep -i "error" log.txt command will return all lines in the log.txt file containing the word "error", case insensitive. And the grep -r "pattern" /var/log command will return all lines from all files in the /var/log directory that contain the pattern "pattern", including files in subdirectories.*

1. Prepare an initial version of the report in electronic form:

* Cover sheet, topic and purpose of the paper
* Glossary of terms
* Answers to p.4.1 and p.4.5 from the preliminary preparation tasks

**Хід роботи**

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

***Готував матеріал студент Заїка С.В.***

1. Опрацюйте всі приклади команд, що представлені у лабораторних роботах курсу NDG Linux Essentials - Lab 9: Archiving and Compression та Lab 10: Working With Text. Створіть таблицю для опису цих команд\*\*\*

|  |  |
| --- | --- |
| Назва команди | Її призначення та функціональність |
| mkdir mybackups | Create a new mybackups directory in the user's home directory |
| tar -cvf mybackups/udev.tar /etc/ude | The tar command is used to combine several files into a single file. In this case, the contents of the /etc/udev directory will be saved to the udev.tar archive in the mybackups directory. The -c option tells the tar command to create a tar file. The -v option stands for "verbose," which tells the tar command to show what it is doing. The -f option is used to specify the name of the tar file. |
| tar –tvf mybackups/udev.tar | Display the contents of a tar file using the available parameters (t = list contents, v = detailed, f = file name): |
| cd mybackups ls tar –xvf udev.tar.gz ls  ls etc ls etc/udev ls etc/udev/rules.d | If you want the files to "return" to their original location, you can first navigate to the / directory and then run the tar command. However, in this example, you will need to be logged in as administrator because only administrators can create files in the /etc directory. |
| tar -rvf udev.tar /etc/hosts tar –tvf udev.tar | To add a file to an existing archive, use the -r option of the tar command. |
| ls -l words xz words ls -l words.xz | Using xz and unxz to compress and decompress a file |
| zip words.zip words ls -l words.zip | Use the zip command to compress a word file: |
| mv | Command to move files and folders. |
| chmod | Command to change access rights to files and folders. |
| grep | A command to search for text in a file. |
| df | Command to display information about available file systems. |
| ssh | Command to establish a secure connection between two computers. |
| ls -l words  bzip2 words  ls -l words.bz2 | ls -l words - displays a detailed list of files in the current directory, including permissions, owner, size, and date of creation of the file named words.  bzip2 words - compresses the words file using the Bzip2 algorithm.  ls -l words.bz2 - displays a detailed list of files in the current directory, including permissions, owner, size, and date of creation of the compressed file named words.bz2. |
| ls -l words.bz2  bunzip2 words.bz2  ls -l words | ls -l words.bz2 - displays a detailed list of files in the current directory, including access rights, owner, size and date of creation of the compressed file named words.bz2.  bunzip2 words.bz2 - unzips the compressed file words.bz2 using the Bzip2 algorithm. When this command is executed, it creates an original file named words that contains the content that was compressed into the compressed file words.bz2.  ls -l words - displays a detailed list of the files in the current directory, including permissions, owner, size, and date of creation of the original file named words. This file has the same size and contents as before it was compressed with the bzip2 command, as it was extracted from the compressed words.bz2 file. |
| ls -l words.xz  unxz words.xz  ls -l words | ls -l words.xz - displays a detailed list of files in the current directory, including access rights, owner, size, and date of creation of the compressed file named words.xz.  unxz words.xz - decompresses the compressed file words.xz using the XZ algorithm. When this command is executed, it creates an original file named words that contains the content that was compressed into the compressed words.xz file.  ls -l words - displays a detailed list of files in the current directory, including permissions, owner, size, and date of creation of the original file named words. This file has the same size and contents as before it was compressed with the xz command, as it was unzipped from the compressed words.xz file. |
| zip -r udev.zip /etc/udev | The zip -r udev.zip /etc/udev command creates a compressed archive file named udev.zip that contains all the files and subdirectories in the /etc/udev path. The -r option specifies that the command should create the archive with subdirectories recursively. |
| ls -l udev.zip | ls -l udev.zip - displays a detailed list of files in the current directory, including access rights, owner, size and date of creation of the compressed file named udev.zip. |
| unzip -l udev.zip | The unzip -l udev.zip command displays a list of files in the udev.zip archive, their sizes and dates of creation/modification. This allows you to check the contents of the archive without unpacking it completely. |

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

***Готував матеріал студент Кресан Руслан***

1. Ознайомтесь з командою tar та за її допомогою виконати у терміналі наступні дії:

* **create a file with the .tar extension;**

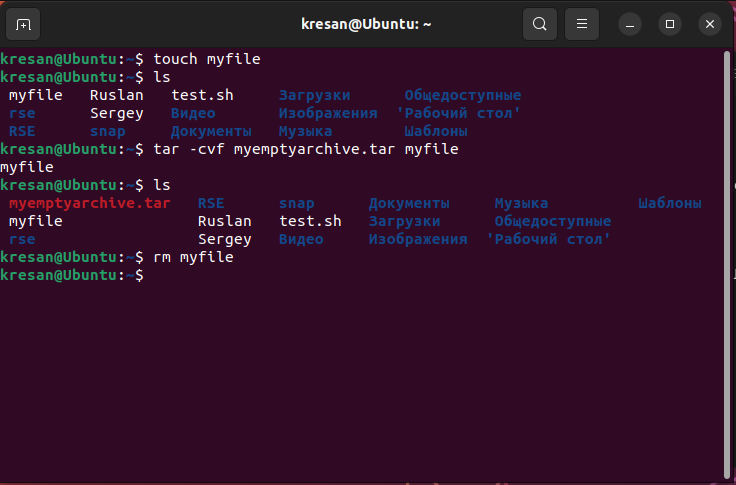
*To create a file with the .tar extension, use the following command:*

***tar -cvf myemptyarchive.tar***

*In my case, we immediately add the file to the archive* ***myfile****:*

***tar -cvf myemptyarchive.tar myfile***

*Because without adding the file, the console displays a message: "tar: Failed to create empty archive ”.*

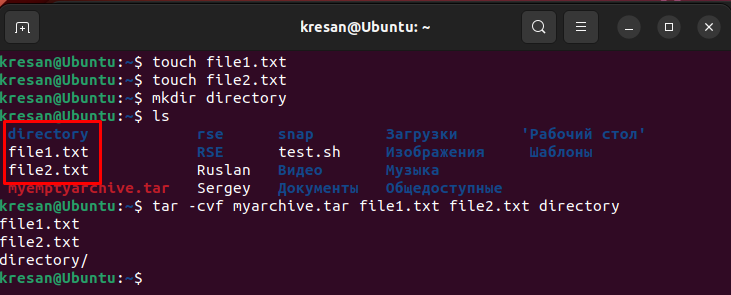


* **create a file with the .tar extension, consisting of several files and directories at the same time;**

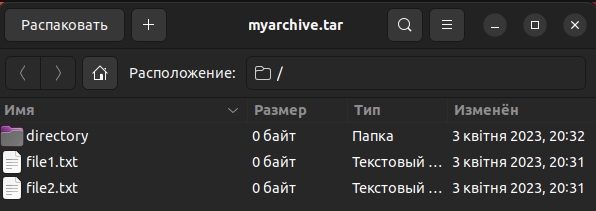
*To execute this command, we created two txt files and one directory. After that, we used the command:*

***Tar -cvf myarchive.tar file1.txt file2.txt directory***

*Which saves the files we created to a file with the .tar extension*.



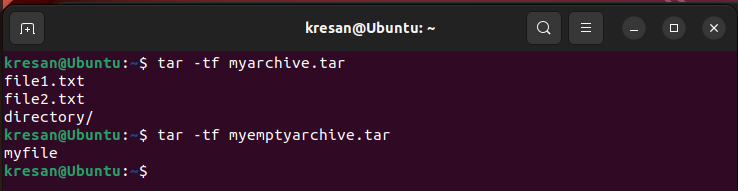
*When we have done these steps, we can open our archive and see the files we have added to it.*



* **view the contents of the file;**

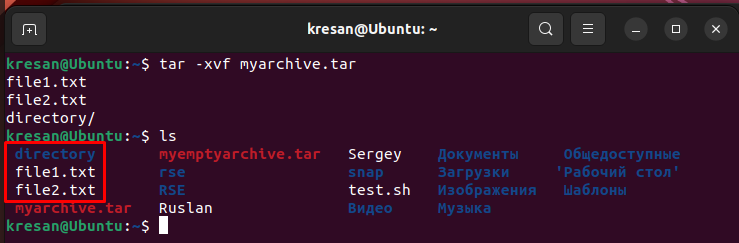
*To view the contents of a file with the tar extension, use the following command:*

***tar -tf «Назва архіву»***

*To demonstrate this, let's look at the contents of the archives we have created.*

* **extract the contents of the tar file;**

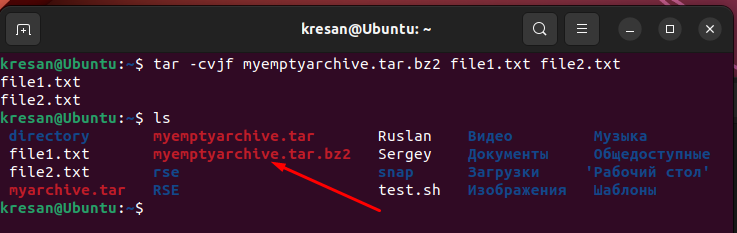
*To extract the contents of a file with the .tar extension, we used the tar command with the -xvf option and specified the name of the archive.*



*After that, the files we extracted from the archive will appear in the current directory.*

* **create a bzip compressed tar archive file;**

*To create an archive file with the .tar extension compressed using bzip, we used the tar command with the -cvjf option, specifying the name of the compressed file and the list of files to be added to the archive.*

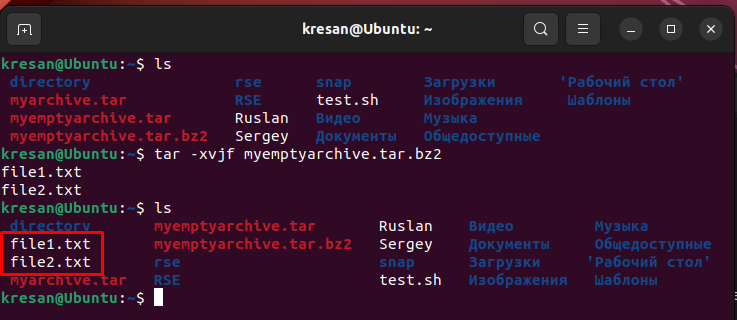


* **extract contents of tar bzip file;**

*To extract the contents of the tar. Bzip file, you can use the command*:

***tar -xvjf myemptyarchive.tar.bz2***

*Where* ***myemptyarchive.tar.bz2*** *is the name of the archive.*

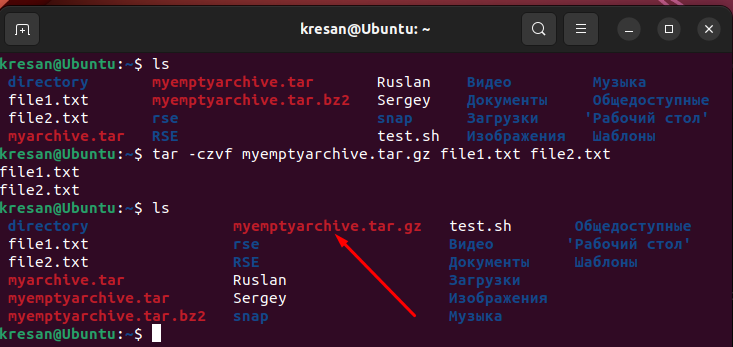


* **create an archive tar file compressed using gzip;**

*To create an archive tar file compressed with gzip, you need to use the command:*

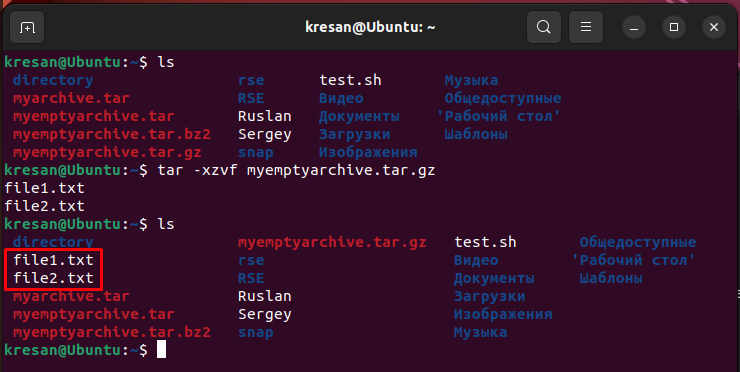
***tar -czvf myarchive.tar.gz file1.txt file2.txt***

*Where* ***myarchive.tar*** *is the name of the archive, and* ***file1.txt*** *та* ***file2.txt*** *are the files*



* **витягти вміст файлу tar gzip.**

*To extract the contents of a tar archive file compressed with gzip, use the tar command with the -xzvf option and the archive name.*



***Готував матеріал студент Кресан Руслан***

1. Як буде відбуватись перенаправлення потоків виведення в bash для наступних дій з командами (позначено як cmd) та файлами (позначено як file):

|  |  |
| --- | --- |
| **Команда** | **Що виконує команда?** |
| cmd 1> file | The command redirects the standard output (STDOUT) from the "cmd" command to a file named "file", overwriting the previous contents of the file. |
| cmd > file | The command has the same effect as the previous one, but it does not specify which output stream is redirected, so it redirects STDOUT by default. |
| cmd 2> file | The command redirects the error stream output (STDERR) from the "cmd" command to a file named "file", overwriting the previous contents of the file. |
| cmd >> file | The command redirects the standard output from the "cmd" command to a file named "file", but it appends the contents to the end of the file instead of overwriting. |
| cmd &> file | The command redirects both STDOUT and STDERR from the "cmd" command to a file named "file", overwriting the previous contents of the file. |
| cmd > file 2>&1 | The command redirects STDOUT from the "cmd" command to a file named "file" and also redirects STDERR to the same file using the combined redirection operator "&>". |
| cmd >> file 2>&1 | The command redirects STDOUT from the "cmd" command to a file named "file" and also redirects STDERR to the same file using the combined redirection operator "&>" and appends the contents to the end of the file. |
| cmd 2>&1 > /dev/null | The command redirects STDERR to STDOUT, meaning that both output streams are redirected to the same location. Next, STDOUT is redirected to /dev/null, which is a special file that discards content. |
| cmd 2> /dev/null | The command redirects the error stream (STDERR) from the "cmd" command to /dev/null, which allows you to discard all error messages. |
| cmd1 | cmd2 | The command passes the output of the standard input stream (STDIN) from the "cmd1" command to the "cmd2" command as standard output (STDOUT). That is, the "cmd2" command receives input from "cmd1" and processes it. |
| cmd1 2>&1 | cmd2 | The command redirects the error stream (STDERR) from the "cmd1" command to STDOUT, and then passes it to the "cmd2" command through the standard output (STDOUT). That is, the cmd2 command receives both standard input and the error stream (STDERR) from cmd1 and processes them. |

***Готував матеріал студент Губенко Є.О***

1. Розгляньте наведені нижче приклади та поясніть, що виконують дані команди та який тип перенаправлення потоків вони використовують:

|  |  |  |
| --- | --- | --- |
| **Команда**  **(контейнер команд)** | **Що виконує команда?** | **Який потік перенаправлення?** |
| $echo "It is a new story." > story | This command writes the string "It is a new story." to a file named "story" using the redirect operator ">". | The stream that is redirected is the standard output (STDOUT), since it is the one that outputs the result of the "echo" command. The ">" operator redirects the output of the command to the "story" file, overwriting any previous contents of the file. |
| $ date > date.txt | This command writes the current date and time to a file named "date.txt" using the ">" redirect operator. | The stream that is redirected is the standard output (STDOUT), since the "date" command outputs the result of its work to this stream. The ">" operator redirects the command's output to the "date.txt" file, overwriting any previous file content, or creating a new file if it does not exist. |
| $ cat file1 file2 file3 > bigfile | This command merges the contents of three files "file1", "file2", and "file3" into a single file "bigfile" using the ">" redirect operator. | The redirected stream is the standard output (STDOUT). When the "cat" command is executed without any command line arguments, it expects data to be entered into the standard input (STDIN) of the user terminal, but in this case, due to the presence of arguments - file names, it reads the contents of these files and writes them to the standard output. The ">" operator redirects the contents of the standard output of the "cat" command to the "bigfile" file, overwriting any previous contents of the file, or creating a new file if it does not exist. |
| $ls -l >> directory | This command appends the output of the "ls -l" command to the end of a file named "directory" using the ">>" redirect operator. | The stream that is redirected is the standard output (STDOUT), because the "ls -l" command prints a list of files with detailed information about them to the standard output. The ">>" operator redirects the command output to the end of the "directory" file, adding new data to it. If the file does not exist, it will be created. |
| $ sort < file1\_unsorted > file2\_sorted | This command sorts the contents of the file "file1\_unsorted" and writes the sorted result to a file named "file2\_sorted" using the "<" and ">" redirection operators. | The stream that is redirected is the standard input (STDIN) stream because the "sort" command expects data to be entered on this stream. The "<" operator redirects the contents of the file "file1\_unsorted" to the standard input of the "sort" command. Then, the "sort" command sorts the entered data and outputs the result to the standard output (STDOUT). The ">" operator redirects the output of the "sort" command to the "file2\_sorted" file, overwriting any previous contents of the file, or creating a new file if it does not exist. |
| $ find -name '\*.txt' > file.txt 2> /dev/null | This command searches for files with the extension ".txt" in the current directory and all its subdirectories using the "find" command, and writes the list of found files to a file named "file.txt" using the redirect operators ">" and "2> /dev/null". | The stream that is redirected from the "find" command is the standard error output (STDERR), since errors may occur during the execution of the command and will be output to this stream. The "2>" operator redirects the error output to a special file device "/dev/null", which does not store the contents of what is directed to it. This means that any errors encountered while executing the "find" command will be ignored. The ">" operator redirects the standard output of the "find" command to the "file.txt" file, overwriting any previous file contents, or creating a new file if it does not exist. |
| $ cat file1\_unsorted | sort > file2\_sorted | This command sorts the contents of the file "file1\_unsorted" and writes the sorted result to a file named "file2\_sorted" using the "|" and ">" redirection operators. | The stream that is redirected is the standard output (STDOUT), since the "cat" command outputs the contents of the file "file1\_unsorted" to this stream, and the "sort" command expects data to be input to this stream. The "|" (pipe) operator redirects the output of the "cat" command to the input of the "sort" command. Then, the "sort" command sorts the entered data and outputs the result to the standard output (STDOUT). The ">" operator redirects the output of the "sort" command to the "file2\_sorted" file, overwriting any previous contents of the file, or creating a new file if it does not exist. |
| $ cat myfile | grep student | wc -l | This command reads the contents of the file "myfile" and applies the "grep" command with the argument "student" to it, which finds all lines containing the word "student". The result is then passed to the "wc -l" command, which counts the number of lines in the input text and outputs the number of lines found to the standard output (STDOUT). | So, this command searches for the number of lines containing the word "student" in the file "myfile" and prints this number to the standard output. The stream that is redirected is the standard input (STDIN) because the "cat" command outputs the contents of the "myfile" file to this stream for further processing by the "grep" and "wc" commands. |

***Готував матеріал студент Губенко Є.О***

**Відповіді на контрольні запитання**

1. Надайте порівняльну характеристику процесам стискання та архівування.

*The compression process and the archiving process are two different methods of reducing the amount of data to be stored or transferred.*

*Data compression is used to reduce the size of a file by removing repetitive data from the original file and replacing it with short pointers that refer to previously used data. This allows you to save more data on the same disk space or reduce the amount of data transfer over the network. If a file is compressed, a decompression procedure is required to restore it. Examples of compression programs include gzip, bzip2, lzma, zstd.*

*On the other hand, data archiving is used to combine multiple files into a single file or to save data as an archive that can contain multiple files in a compressed form, preserving their names and other information. This is convenient for saving or transferring multiple files at once. When archiving, information about file names and their structure is saved. To unzip files from an archive, you need to perform an unzipping procedure. Examples of archiving programs include tar, zip, rar, 7zip.*

*So, although both methods are used to reduce the amount of data, their use depends on the specific situation. If you need to reduce the size of a single file, then compression is the best option. Archiving is more convenient if you need to transfer multiple files or preserve the directory structure.*

1. Які програми, окрім наведених в роботі, можуть використовуватись для стискання та архівування файлів та каталогів в ОС Linux? Наведіть приклади та їх короткий опис.

*Linux has many programs for compressing and archiving files and directories. Some of them are:*

* ***zip:*** *a program for creating and unpacking ZIP archives. It uses the Deflate compression algorithm to reduce the size of files.*
* ***unzip:*** *a program for unpacking ZIP archives.*
* ***rar:*** *a program for creating and unpacking RAR archives. It uses sophisticated compression algorithms to reduce the size of files.*
* ***unrar:*** *a program for unpacking RAR archives.*
* ***tar:*** *a program for creating and unpacking tar archives. It does not compress files, but only combines them into a single archive file.*
* ***gzip:*** *a program for compressing files using the gzip algorithm. It reduces the size of files by replacing repeated sequences of characters with references to previous occurrences of these sequences in the file.*
* ***unzip:*** *a program for unpacking gzip-compressed files.*
* ***bzip2:*** *a program for compressing files using the bzip2 algorithm. It uses a more sophisticated compression algorithm that provides a higher compression ratio, but also requires more computer resources.*
* ***bunzip2:*** *a program for unzipping bzip2-compressed files.*

*These programs can be useful when compressing and archiving files and directories on Linux.*

1. Порівняйте алгоритми стискання, що використовуються в командах (програмах), використовуваних в Linux. Які з алгоритмів можна вважати найшвидшим та найефективнішим?

*There are several compression algorithms available on Linux that are used in various programs, such as gzip, bzip2, xz, and tar programs that can use these algorithms to create archives.*

*The gzip algorithm uses the DEFLATE compression algorithm, which is very fast and uses a small amount of RAM. However, it does not provide the highest compression ratio, and therefore a file compressed with gzip may be slightly larger than the same file compressed with other algorithms.*

*The bzip2 algorithm uses more RAM and CPU resources than gzip, but usually gives a better compression ratio. Therefore, if the goal is to maximize file compression, it is better to use bzip2.*

*The xz algorithm uses even more CPU and RAM resources than bzip2, and therefore usually gives an even better compression ratio. But because of its high resource consumption, it may run slower on less powerful computers.*

*Since gzip is faster and uses fewer resources, it is usually the fastest option for quickly compressing files that do not require the maximum compression ratio. However, if you need to reduce the file size as much as possible, then using bzip2 or xz.*

1. Опишіть програмні засоби для стискання та архівування, що можуть бути використані у вашому мобільному телефоні.

*Most modern cell phones have built-in software tools for compressing and archiving files. Here are some of them:*

* ***Google Files*** *- is a file management program that also has the ability to compress and archive files. It can compress files in ZIP, 7Z formats, and create a ZIP archive from several files.*
* ***WinZip*** *- is a popular program for archiving and compressing files. It is available for many mobile platforms such as Android and iOS. WinZip allows you to create ZIP archives, unzip archives and compress files in ZIP and RAR formats.*
* ***RAR*** *- is a file compression and archiving application available for Android and iOS. It allows you to create RAR and ZIP archives, unzip archives, and compress files in ZIP, RAR, and 7Z formats..*
* ***ZArchiver*** *- is an Android application that allows you to compress and decompress files in ZIP, 7Z, TAR, GZIP, and BZIP2 formats. It also supports file encryption and can act as a file manager.*
* ***iZip*** *- is an iOS app that allows you to compress and decompress files in ZIP, RAR, 7Z, TAR, GZIP, and BZIP2 formats. It also has the ability to create ZIP archives from multiple files and supports file manager functions.*

*These programs are very useful for storing and transferring files, especially if the file exchange takes place on a mobile device.*

1. Опишіть та порівняйте програмні засоби для стискання та (де)архівування даних у ОС сімейства Windows.

*There are many software tools for compressing and archiving data in the Windows family of operating systems. The most common ones are:*

* ***Windows built-in zip****: is a Windows-based tool for archiving and compressing files and directories. It supports compression in zip, gzip, and tar formats. Windows built-in zip is easy to use and supports encryption.*
* ***WinZip****: is one of the most popular tools for archiving and compressing data on Windows. WinZip supports a wide range of archive formats, including zip, gzip, tar, RAR and others. It also has the ability to encrypt and password protect.*
* ***7-Zip****: is a free and open-source program that supports compression in 7z, ZIP, GZIP, BZIP2 and TAR formats. 7-Zip has a high compression ratio and speed.*
* ***WinRAR****: is a commercial data archiving and compression tool that supports a wide range of archive formats, including RAR, ZIP, CAB, ARJ, LZH, TAR, GZip, UUE, ISO, and others. WinRAR has the ability to encrypt and protect with a password.*

*Compared to the built-in Windows archiving tool, WinZip, 7-Zip and WinRAR have more features and a wider range of supported archive formats. In terms of performance and compression ratio, 7-Zip is generally considered to have one of the fastest and most efficient compression algorithms compared to other programs.*

1. Поясніть яким чином стиснення та архівування даних може бути використано для резервування даних. В яких ще задачах системного адміністрування воно може бути використано.

*Data compression and archiving can be used for data backup, i.e. to save a copy of important data in a safe place for later recovery in case of loss or damage to the main data.*

*Data backup usually involves using data archiving and compression programs that allow you to combine many files and directories into a single archive file, reducing the amount of data and making it easier to store on external media.*

*After archiving and compressing the data, the backup file can be copied to an external storage device such as a USB flash drive, external hard drive, or cloud storage, where it will be stored safely until you need to restore it.*

*In addition to backup, data compression and archiving can be used to efficiently store large amounts of data in limited storage space, to transfer data over a network with less bandwidth usage, and to efficiently organize and store data in systems with limited resources.*

1. Яке призначення директорії файлу /dev/null?

*The /dev/null directory in Linux and other Unix-like operating systems is used to send data to it and discard it without saving it. It is not a regular directory containing files, but a ghost file that is available as a device on the system. When data is written to /dev/null, it will be completely discarded and not saved on the system. This can be useful for programs that generate output that is not needed, or for redirecting program output to nowhere.*

*For example, if you want to execute a command but don't want to save the output, you can direct the output to /dev/null instead of saving it to a file: $ some\_command > /dev/null*

*Also, /dev/null can be used to destroy confidential information, as the data written to this file cannot be recovered.*

***Готував матеріал студент Губенко Є.О***

**Висновки**

During the lab, we gained skills in working with the Bash shell and got acquainted with basic commands for archiving and compressing data and basic actions when working with text in the terminal.