**ЛАБОРАТОРНА РОБОТА № 2**

**Виконання дій у системі за допомогою скриптів**

**Хід роботи:**

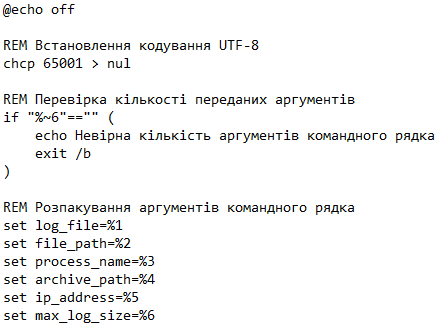
Посилання на репозиторій:

<https://github.com/andreylion06/SystemNetworkProgramming-Lab2>

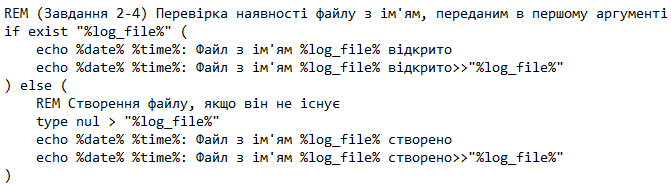
***Завдання:***

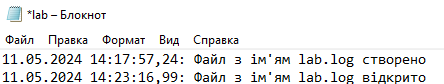
Написати пакетний файл який:

1. Має аргументи командного рядка: *Аргумент1, Аргумент2, Аргумент3, Аргумент4, Аргумент5, Аргумент6.*Приклад виклику(і це тільки приклад, ваш виклик буде виглядати по іншому:***lab.cmd lab.log d:\baza xx.exe d:\archive 192.168.1.2 500***де:***lab.cmd****-*назва вашого файла скрипта ***lab.log****- назва вашого log файлу****d:\baza****-*шлях до файлів які підлягають регулярному архівуванню***xx.exe -***назва процесу який потрібно завершити***d:\archive****-*шлях, по якому необхідно записати архів***192.168.1.2****-*IP адреса комп'ютера, який потрібно вимкнути***500****-*розмір log файлу, який не має бути перевищеним.

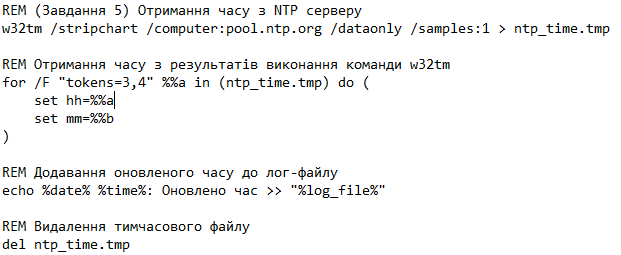


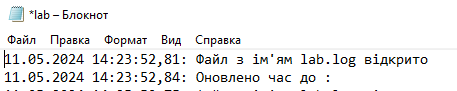
1. Перевіряє чи існує файл, ім’я якого завдано у *Аргумент1*.
2. Якщо не існує, то створює його. Це буде log файл скрипта.
3. Дописує у цей файл :
   * поточну дату та час;
   * «Файл з ім’ям *Аргумент1* відкрито або створено».



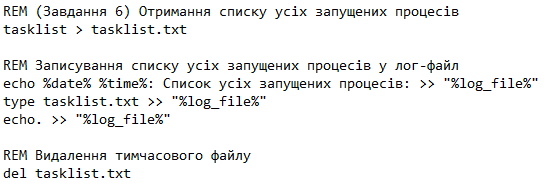


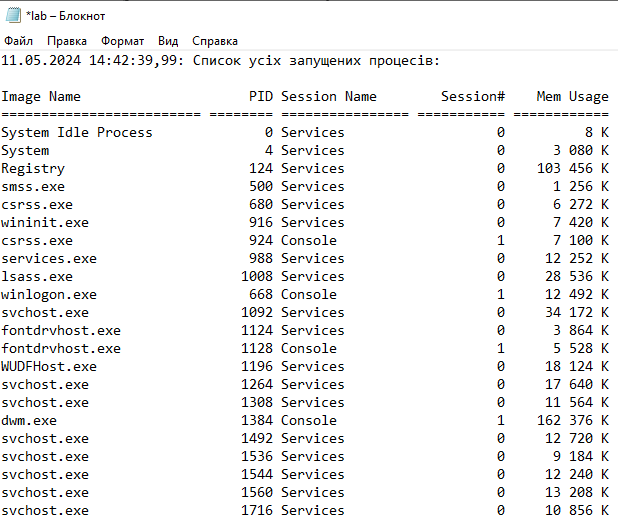
1. Отримати час з NTP серверу та встановити його  поточним. Записати оновлений час у log.



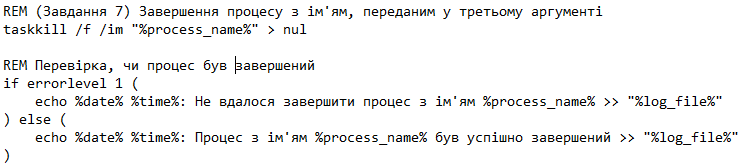


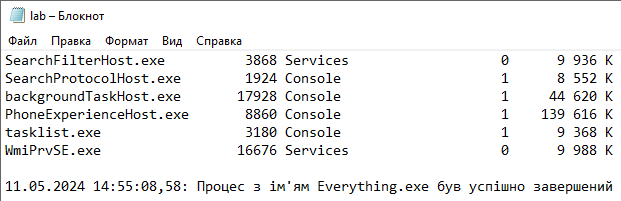
1. Виводить у log список усіх запущених процесів.





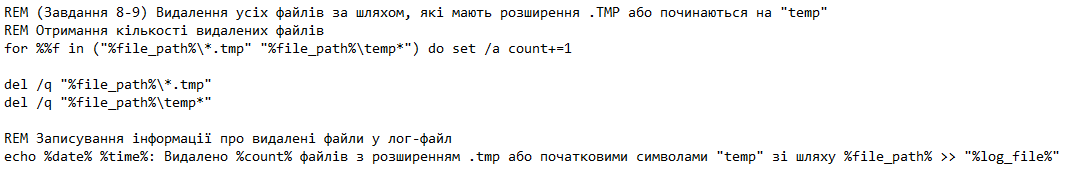
1. Завершує процес, з ім’ям *Аргумент3*. Інформує в log.



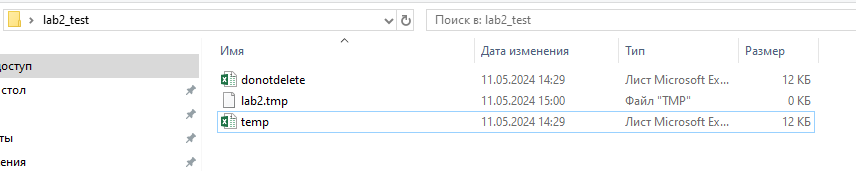


В диспетчері задач процес більше не відображається.

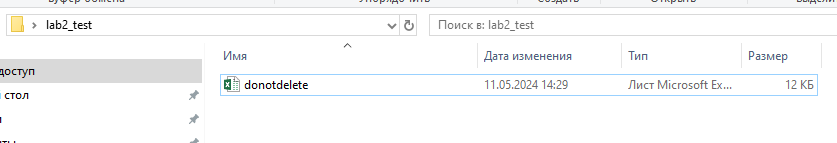
1. Видаляє усі файли за шляхом *Аргумент2*, які мають розширення .TMP, або їх ім’я починається на «temp».
2. Інформацію про виконані дії записує у log файл. Вказати кількість видалених файлів.

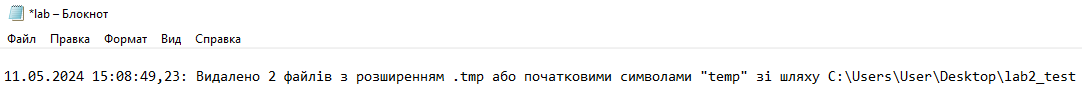


Папка, шлях якої я вказав у скрипті до його виконання:

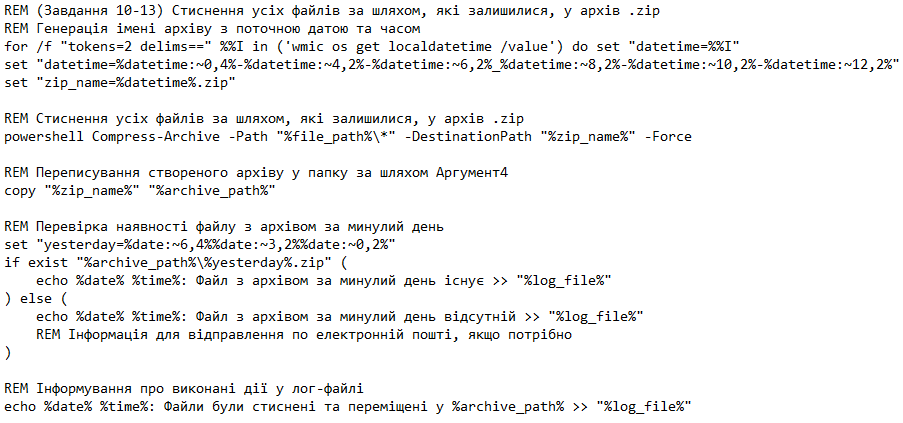


Папка після виконання скрипта:

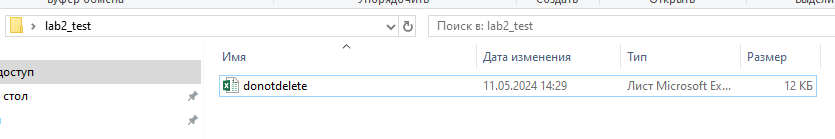




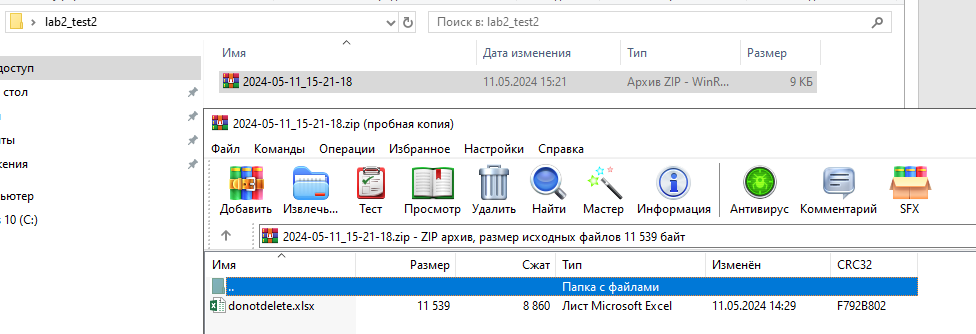
1. Стискає усі файли які залишилися за шляхом *Аргумент2*у архів .zip. Ім’я архіву – поточна дата та час.
2. Переписує створений архів у папку за шляхом *Аргумент4*.
3. Перевіряє чи є файл з архівом за минулий день. Інформує у log.Якщо нема, інформує на email.
4. Інформацію про виконані дії записує у log файл.



Папка для архівування

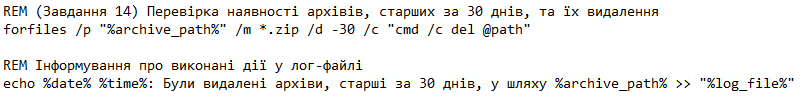


Заархівована папка за вказаним шляхом



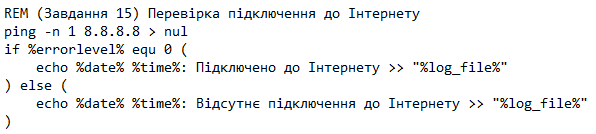


1. Перевіряє, чи є за шляхом *Аргумент4*, архіви, старші 30 днів, та якщо є, то видаляє. Інформує в log.



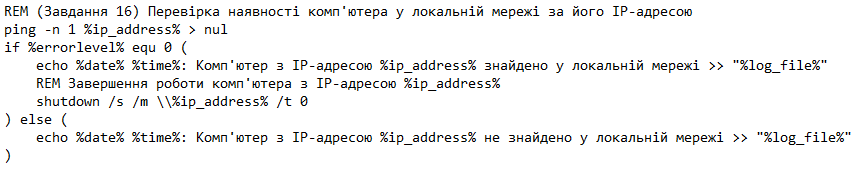


1. Перевіряє чи є підключення до Internet, та інформує в log.



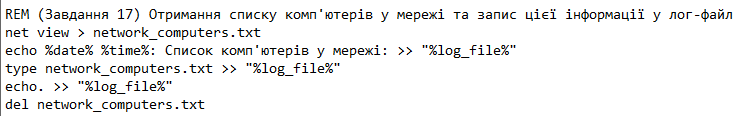


1. Перевірити чи є в локальній мережі комп’ютер з IP *аргумент5*, та якщо є, то завершити його роботу. Інформувати в log.

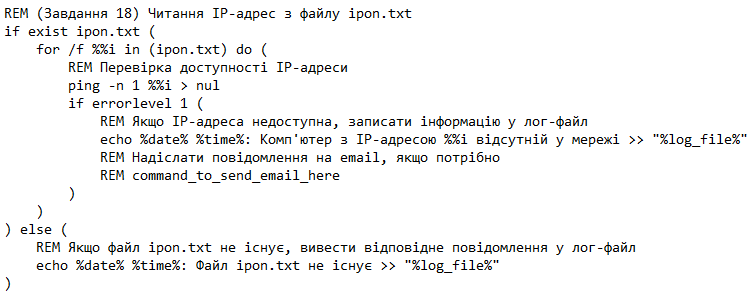




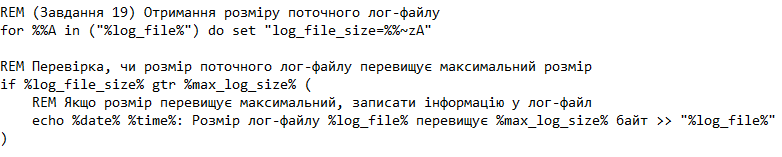
1. Отримує список комп’ютерів в мережі та записує отриману інформацію у log.



1. Перевіряє, якщо один з комп’ютерів, зі вказаними IP адресами у файлі ipon.txt не присутній в мережі, інформує в log та на email.

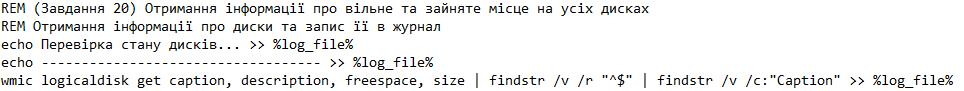


1. Перевіряє, якщо розмір поточного log файлу більший за *Аргумент6*, то інформує в log та на email.



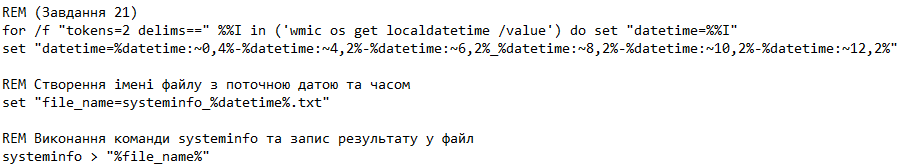


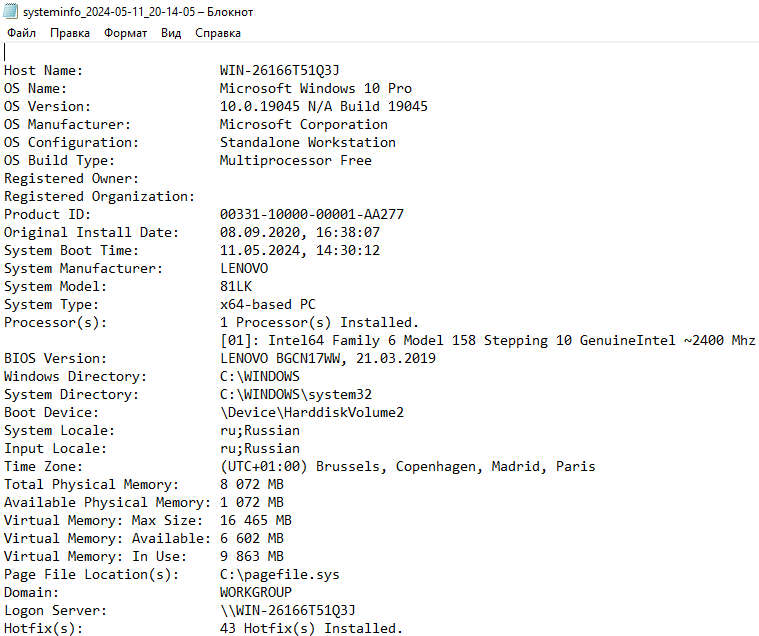
1. Перевіряє кількість вільного та зайнятого місця на усіх дисках в системі та пише цю інформацію в log.



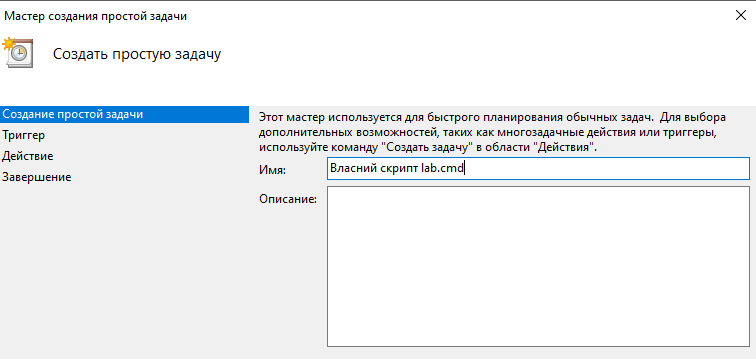


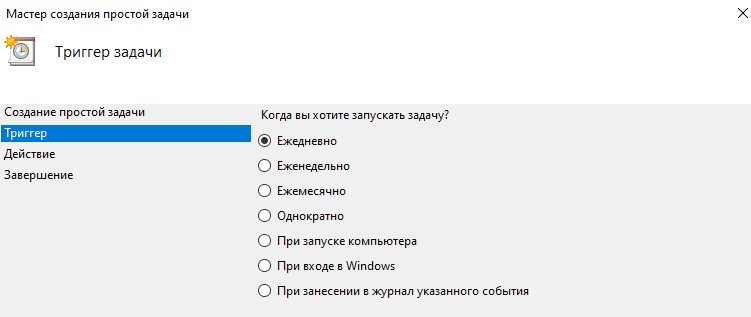
1. Записати результат виконання команди systeminfo у файл “*systeminfo+поточна дата-час.txt*”.

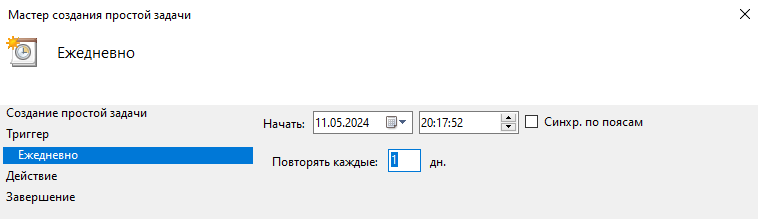


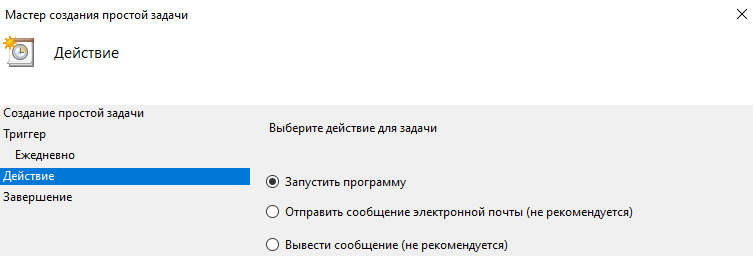


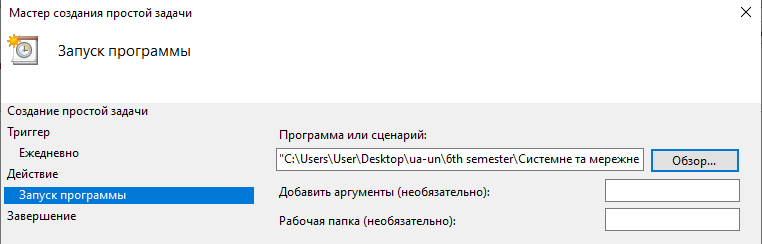
1. Встановіть запуск вашого скрипту, регулярно, періодично.











1. Описати у звіті кожну використану внутрішню та зовнішню команду зі всіма параметрами.

chcp 65001 > nul

На початку файлу встановлюю кодування UTD-8 для підтримки символів Юнікоду у виведені в консоль.

if "%~6"=="" (

echo Невірна кількість аргументів командного рядка

exit /b

)

Перевірка кількості переданих аргументів. Після чого відбуваєтсья розпаковка аргументів командного рядка та присвоєння значення змінним для подальшого використання.

Далі вже відбувається саме виконання завдань, логіка та використання команд гарно висвітлюється за допомогою коментарів у лістингу програми.

Пакетний файл я викликаю з папки його знаходження з наступною командою:

**lab.cmd lab.log C:\Users\User\Desktop\lab2\_test Everything.exe C:\Users\User\Desktop\lab2\_test2 192.168.1.2 50**

***Лістинг програми:***

@echo off

REM Встановлення кодування UTF-8

chcp 65001 > nul

REM Перевірка кількості переданих аргументів

if "%~6"=="" (

echo Невірна кількість аргументів командного рядка

exit /b

)

REM Розпакування аргументів командного рядка

set log\_file=%1

set file\_path=%2

set process\_name=%3

set archive\_path=%4

set ip\_address=%5

set max\_log\_size=%6

REM (Завдання 2-4) Перевірка наявності файлу з ім'ям, переданим в першому аргументі

if exist "%log\_file%" (

echo %date% %time%: Файл з ім'ям %log\_file% відкрито

echo %date% %time%: Файл з ім'ям %log\_file% відкрито>>"%log\_file%"

) else (

REM Створення файлу, якщо він не існує

type nul > "%log\_file%"

echo %date% %time%: Файл з ім'ям %log\_file% створено

echo %date% %time%: Файл з ім'ям %log\_file% створено>>"%log\_file%"

)

REM (Завдання 5) Отримання часу з NTP серверу

w32tm /stripchart /computer:pool.ntp.org /dataonly /samples:1 > ntp\_time.tmp

REM Отримання часу з результатів виконання команди w32tm

for /F "tokens=3,4" %%a in (ntp\_time.tmp) do (

set hh=%%a

set mm=%%b

)

REM Додавання оновленого часу до лог-файлу

echo %date% %time%: Оновлено час >> "%log\_file%"

REM Видалення тимчасового файлу

del ntp\_time.tmp

REM (Завдання 6) Отримання списку усіх запущених процесів

tasklist > tasklist.txt

REM Записування списку усіх запущених процесів у лог-файл

echo %date% %time%: Список усіх запущених процесів: >> "%log\_file%"

type tasklist.txt >> "%log\_file%"

echo. >> "%log\_file%"

REM Видалення тимчасового файлу

del tasklist.txt

REM (Завдання 7) Завершення процесу з ім'ям, переданим у третьому аргументі

taskkill /f /im "%process\_name%" > nul

REM Перевірка, чи процес був завершений

if errorlevel 1 (

echo %date% %time%: Не вдалося завершити процес з ім'ям %process\_name% >> "%log\_file%"

) else (

echo %date% %time%: Процес з ім'ям %process\_name% був успішно завершений >> "%log\_file%"

)

REM (Завдання 8-9) Видалення усіх файлів за шляхом, які мають розширення .TMP або починаються на "temp"

REM Отримання кількості видалених файлів

for %%f in ("%file\_path%\\*.tmp" "%file\_path%\temp\*") do set /a count+=1

del /q "%file\_path%\\*.tmp"

del /q "%file\_path%\temp\*"

REM Записування інформації про видалені файли у лог-файл

echo %date% %time%: Видалено %count% файлів з розширенням .tmp або початковими символами "temp" зі шляху %file\_path% >> "%log\_file%"

EM (Завдання 10-13) Стиснення усіх файлів за шляхом, які залишилися, у архів .zip

REM Генерація імені архіву з поточною датою та часом

for /f "tokens=2 delims==" %%I in ('wmic os get localdatetime /value') do set "datetime=%%I"

set "datetime=%datetime:~0,4%-%datetime:~4,2%-%datetime:~6,2%\_%datetime:~8,2%-%datetime:~10,2%-%datetime:~12,2%"

set "zip\_name=%datetime%.zip"

REM Стиснення усіх файлів за шляхом, які залишилися, у архів .zip

powershell Compress-Archive -Path "%file\_path%\\*" -DestinationPath "%zip\_name%" -Force

REM Переписування створеного архіву у папку за шляхом Аргумент4

copy "%zip\_name%" "%archive\_path%"

REM Переписування створеного архіву у папку за шляхом Аргумент4

copy "%zip\_name%" "%archive\_path%"

REM Перевірка наявності файлу з архівом за минулий день

set "yesterday=%date:~6,4%%date:~3,2%%date:~0,2%"

if exist "%archive\_path%\%yesterday%.zip" (

echo %date% %time%: Файл з архівом за минулий день існує >> "%log\_file%"

) else (

echo %date% %time%: Файл з архівом за минулий день відсутній >> "%log\_file%"

REM Інформація для відправлення по електронній пошті, якщо потрібно

)

REM Інформування про виконані дії у лог-файлі

echo %date% %time%: Файли були стиснені та переміщені у %archive\_path% >> "%log\_file%"

REM (Завдання 14) Перевірка наявності архівів, старших за 30 днів, та їх видалення

forfiles /p "%archive\_path%" /m \*.zip /d -30 /c "cmd /c del @path"

REM Перевірка, чи були видалені архіви

if %errorlevel% equ 0 (

echo %date% %time%: Були видалені архіви, старші за 30 днів, у шляху %archive\_path% >> "%log\_file%"

) else (

echo %date% %time%: Не було знайдено архівів, старших за 30 днів, у шляху %archive\_path% >> "%log\_file%"

)

REM (Завдання 15) Перевірка підключення до Інтернету

ping -n 1 8.8.8.8 > nul

if %errorlevel% equ 0 (

echo %date% %time%: Підключено до Інтернету >> "%log\_file%"

) else (

echo %date% %time%: Відсутнє підключення до Інтернету >> "%log\_file%"

)

REM (Завдання 16) Перевірка наявності комп'ютера у локальній мережі за його IP-адресою

ping -n 1 %ip\_address% > nul

if %errorlevel% equ 0 (

echo %date% %time%: Комп'ютер з IP-адресою %ip\_address% знайдено у локальній мережі >> "%log\_file%"

REM Завершення роботи комп'ютера з IP-адресою %ip\_address%

shutdown /s /m \\%ip\_address% /t 0

) else (

echo %date% %time%: Комп'ютер з IP-адресою %ip\_address% не знайдено у локальній мережі >> "%log\_file%"

)

REM (Завдання 17) Отримання списку комп'ютерів у мережі та запис цієї інформації у лог-файл

net view > network\_computers.txt

echo %date% %time%: Список комп'ютерів у мережі: >> "%log\_file%"

type network\_computers.txt >> "%log\_file%"

echo. >> "%log\_file%"

del network\_computers.txt

REM (Завдання 18) Читання IP-адрес з файлу ipon.txt

if exist ipon.txt (

for /f %%i in (ipon.txt) do (

REM Перевірка доступності IP-адреси

ping -n 1 %%i > nul

if errorlevel 1 (

REM Якщо IP-адреса недоступна, записати інформацію у лог-файл

echo %date% %time%: Комп'ютер з IP-адресою %%i відсутній у мережі >> "%log\_file%"

REM Надіслати повідомлення на email, якщо потрібно

REM command\_to\_send\_email\_here

)

)

) else (

REM Якщо файл ipon.txt не існує, вивести відповідне повідомлення у лог-файл

echo %date% %time%: Файл ipon.txt не існує >> "%log\_file%"

)

REM (Завдання 19) Отримання розміру поточного лог-файлу

for %%A in ("%log\_file%") do set "log\_file\_size=%%~zA"

REM Перевірка, чи розмір поточного лог-файлу перевищує максимальний розмір

if %log\_file\_size% gtr %max\_log\_size% (

REM Якщо розмір перевищує максимальний, записати інформацію у лог-файл

echo %date% %time%: Розмір лог-файлу %log\_file% перевищує %max\_log\_size% байт >> "%log\_file%"

)

REM (Завдання 20) Отримання інформації про вільне та зайняте місце на усіх дисках

REM Отримання інформації про диски та запис її в журнал

echo Перевірка стану дисків... >> %log\_file%

echo ----------------------------------- >> %log\_file%

wmic logicaldisk get caption, description, freespace, size | findstr /v /r "^$" | findstr /v /c:"Caption" >> %log\_file%

REM (Завдання 21)

for /f "tokens=2 delims==" %%I in ('wmic os get localdatetime /value') do set "datetime=%%I"

set "datetime=%datetime:~0,4%-%datetime:~4,2%-%datetime:~6,2%\_%datetime:~8,2%-%datetime:~10,2%-%datetime:~12,2%"

REM Створення імені файлу з поточною датою та часом

set "file\_name=systeminfo\_%datetime%.txt"

REM Виконання команди systeminfo та запис результату у файл

systeminfo > "%file\_name%"

***Висновок:*** В результаті виконання лабораторної роботи був розроблений пакетний файл, який автоматизує ряд завдань з обробки файлів, моніторингу системи та мережі. Використані внутрішні та зовнішні команди дозволили ефективно виконати поставлені завдання, забезпечуючи автоматизацію та зручність управління. Результати роботи пакетного файлу демонструють його потенціал для оптимізації рутинних завдань адміністрування системи.