ЛАБОРАТОРНА РОБОТА № 3

***Тема:* «Створення утіліти “DiskInfo”**».

***Мета роботи:*** ознайомитися з системними функціями, щоб отримати інформацію про дискову підсистему та використати їх на практиці.

**Хід роботи**

***Завдання на лабораторну роботу:***

У даній лабораторній работі, використовуючи виклики системних функцій, отримати інформацію про дискову підсистему:

1. Список усіх логічних дисків в системі.
2. Отримати тип кожного диску присутнього в системі, та дати пояснення для кожного типу диску.
3. Отримати інформацію про диски в системі та про файлові системи, які використовують на них.
4. Отримати інформацію про зайнятості та вільне місце на кожному з дисків.
5. Отримати інформацію про системну пам'ять.
6. Отримати інформацію про назву комп'ютера
7. Отримати назву поточного користувача
8. Отримати інформацію про поточний системний каталог, тимчасовий каталог, поточний робочий каталог.
9. Для обраних каталогу на диску, включити спостереження за змінами, продемонструвати відслідковування більше однієї зміни. Зміни записувати в лог файл.

Для реалізації завдання використаємо мову програмування C# та платформу .NET 6.

Використані пакети, зокрема в пункті 5:

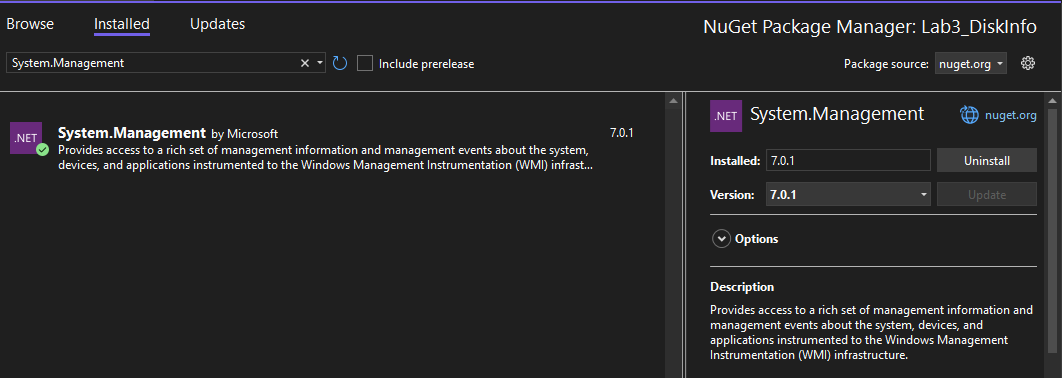


Рис. 1. Використані пакети

Повний лістинг програми:

using System.Management;

using System.Text;

Console.OutputEncoding = Encoding.Unicode;

Console.InputEncoding = Encoding.Unicode;

Console.ForegroundColor = ConsoleColor.Green;

var menu = 0;

do

{

Console.WriteLine("\t\t\t\t\t\t\tПрограма \"DiskInfo\"");

Console.WriteLine("\nМеню:");

Console.WriteLine("1) Список усіх логічних дисків в системі");

Console.WriteLine("2) Тип кожного диску присутнього в системі");

Console.WriteLine("3) Інформація про файлові системи, які використовують диски та про зайняте та вільне місце на кожному з дисків");

Console.WriteLine("4) Інформація про системну пам'ять");

Console.WriteLine("5) Інформація про назву комп'ютера");

Console.WriteLine("6) Назва поточного користувача");

Console.WriteLine("7) Інформація про поточний системний каталог, тимчасовий каталог, поточний робочий каталог");

Console.WriteLine("8) Спостерігати за змінами для обраного каталогу");

Console.Write("\nОберіть пункт меню або 0, щоб завершити роботу програми: ");

bool flag;

do

{

flag = int.TryParse(Console.ReadLine(), out menu);

if (!flag || menu is < 0 or > 8)

{

Console.WriteLine("\nУпс, помилка введення! Перевірте коректність введеного значення та повторіть введення іще раз!");

}

} while (!flag || menu is < 0 or > 8);

var allDrives = DriveInfo.GetDrives();

switch (menu)

{

case 1:

foreach (var drive in allDrives)

{

Console.WriteLine($"Диск {drive.Name}");

}

break;

case 2:

foreach (var drive in allDrives)

{

switch (drive.DriveType)

{

case DriveType.CDRom:

Console.WriteLine($"{drive.Name} - оптичний диск (CD-ROM)");

break;

case DriveType.Fixed:

Console.WriteLine($"{drive.Name} - фіксований диск");

break;

case DriveType.Network:

Console.WriteLine($"{drive.Name} - мережевий диск");

break;

case DriveType.Removable:

Console.WriteLine($"{drive.Name} - знімний диск");

break;

case DriveType.Ram:

Console.WriteLine($"{drive.Name} - RAM диск");

break;

case DriveType.Unknown:

case DriveType.NoRootDirectory:

default:

Console.WriteLine($"{drive.Name} - невідомий тип диску");

break;

}

}

break;

case 3:

foreach (var drive in allDrives)

{

Console.WriteLine($"Диск {drive.Name}");

Console.WriteLine($"Доступний: {(drive.IsReady ? "Так" : "Ні")}");

if (drive.IsReady)

{

Console.WriteLine($"Мітка: {drive.VolumeLabel}");

Console.WriteLine($"Файлова система: {drive.DriveFormat}");

Console.WriteLine($"Вільне місце: {drive.AvailableFreeSpace / 1024 / 1024} МБ");

Console.WriteLine($"Зайняте місце: {drive.TotalSize / 1024 / 1024} МБ\n");

}

}

break;

case 4:

Console.WriteLine($"Операційна система: {Environment.OSVersion}");

var searcher = new ManagementObjectSearcher("SELECT \* FROM Win32\_OperatingSystem");

foreach (var obj in searcher.Get())

{

var maxPhysicalMemory = Convert.ToUInt64(obj["TotalVisibleMemorySize"]);

var freePhysicalMemory = Convert.ToUInt64(obj["FreePhysicalMemory"]);

var freeVirtualMemory = Convert.ToUInt64(obj["FreeVirtualMemory"]);

Console.WriteLine($"Максимальна фізична пам'ять: {maxPhysicalMemory / 1024f} МБ");

Console.WriteLine($"Вільна фізична пам'ять: {freePhysicalMemory / 1024f} МБ");

Console.WriteLine($"Вільна кількість віртуальної пам'яті: {freeVirtualMemory / 1024f} МБ");

}

break;

case 5:

var computerName = Environment.MachineName;

Console.WriteLine($"Назва комп'ютера: {computerName}");

break;

case 6:

var userName = Environment.UserName;

Console.WriteLine($"Назва поточного користувача: {userName}");

break;

case 7:

var systemFolder = Environment.SystemDirectory;

var tempFolder = Environment.GetEnvironmentVariable("TEMP");

var currentFolder = Environment.CurrentDirectory;

Console.WriteLine($"Поточний системний каталог: {systemFolder}");

Console.WriteLine($"Тимчасовий каталог: {tempFolder}");

Console.WriteLine($"Поточний робочий каталог: {currentFolder}");

break;

case 8:

Console.Write("Введіть повний шлях до обраного каталогу для відслідковування змін: ");

string directoryPath;

do

{

directoryPath = @Console.ReadLine();

flag = Directory.Exists(directoryPath);

if (!flag)

{

Console.WriteLine("\nУпс, помилка введення! Перевірте коректність введеного шляху та повторіть введення нижче іще раз!");

}

} while (!flag);

var watcher = new FileSystemWatcher(directoryPath);

watcher.NotifyFilter = NotifyFilters.Attributes | NotifyFilters.CreationTime | NotifyFilters.DirectoryName |

NotifyFilters.FileName | NotifyFilters.LastAccess | NotifyFilters.LastWrite |

NotifyFilters.Security | NotifyFilters.Size;

watcher.Filter = "\*.\*";

watcher.Changed += (\_, e) =>

{

if (!File.Exists("log.txt"))

{

using var createLogFileWriter = File.CreateText("log.txt");

createLogFileWriter.WriteLine($"Лог-файл створено: {DateTime.Now}");

}

Console.WriteLine($"\nФайл або каталог: {e.FullPath} змінено, Час: {DateTime.Now}");

using var writer = new StreamWriter("log.txt", true);

writer.WriteLine($"Файл або каталог: {e.FullPath} змінено, Час: {DateTime.Now}");

};

watcher.Created += (\_, e) =>

{

if (!File.Exists("log.txt"))

{

using var createLogFileWriter = File.CreateText("log.txt");

createLogFileWriter.WriteLine($"Лог-файл створено: {DateTime.Now}");

}

Console.WriteLine($"\nФайл або каталог: {e.FullPath} створено, Час: {DateTime.Now}");

using var writer = new StreamWriter("log.txt", true);

writer.WriteLine($"Файл або каталог: {e.FullPath} створено, Час: {DateTime.Now}");

};

watcher.Renamed += (\_, e) =>

{

if (!File.Exists("log.txt"))

{

using var createLogFileWriter = File.CreateText("log.txt");

createLogFileWriter.WriteLine($"Лог-файл створено: {DateTime.Now}");

}

Console.WriteLine($"\nФайл або каталог: {e.OldFullPath} -> {e.FullPath} перейменовано, Час: {DateTime.Now}");

using var writer = new StreamWriter("log.txt", true);

writer.WriteLine($"Файл або каталог: {e.OldFullPath} -> {e.FullPath} перейменовано, Час: {DateTime.Now}");

};

watcher.Deleted += (\_, e) =>

{

if (!File.Exists("log.txt"))

{

using var createLogFileWriter = File.CreateText("log.txt");

createLogFileWriter.WriteLine($"Лог-файл створено: {DateTime.Now}");

}

Console.WriteLine($"\nФайл або каталог: {e.FullPath} видалено, Час: {DateTime.Now}");

using var writer = new StreamWriter("log.txt", true);

writer.WriteLine($"Файл або каталог: {e.FullPath} видалено, Час: {DateTime.Now}");

};

watcher.IncludeSubdirectories = true;

watcher.EnableRaisingEvents = true;

Console.WriteLine("\nСпостереження за змінами включено. Натисніть будь-яку клавішу для виходу.");

Console.ReadKey();

break;

}

Console.WriteLine("\nНатисніть на будь-яку кнопку, щоб продовжити");

Console.ReadKey();

Console.Clear();

} while (menu != 0);

Результат виконання програми:

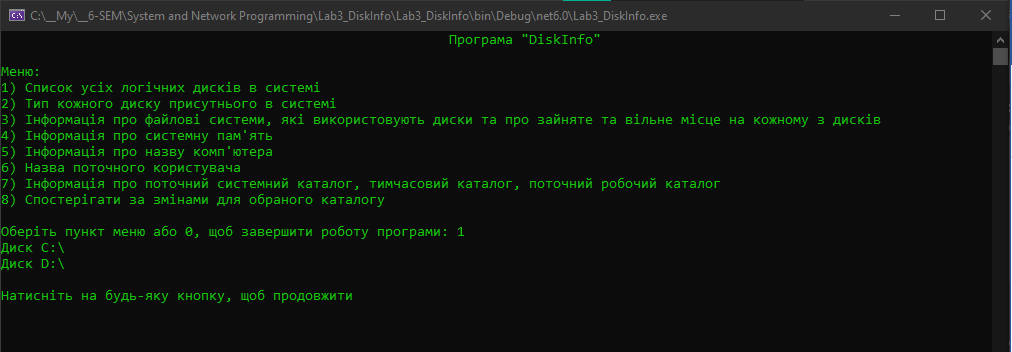


Рис. 2.1. Список усіх логічних дисків в системі та головне меню

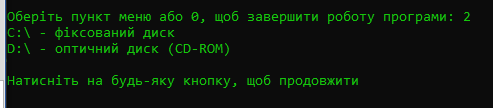


Рис. 2.2. Тип кожного диску присутнього в системі

***Опис кожного типу дисків***

**Фіксований диск** — це накопичувач на жорсткому диску, який залишається встановленим в системі незалежно від наявності фізичної наявності пристрою. Такі диски зазвичай використовуються для зберігання файлів, які потрібні для роботи програм.

**CD-ROM** — компакт-диск з програмним забезпеченням або даними, який можна читати, але не можна записувати. Цей формат диска використовується для зберігання музики, фільмів, програмного забезпечення, даних тощо. CD-ROM диск може бути витягнутий з комп'ютера, тому він не є фіксованим диском.

Мержевий диск — це комбінація твердотільного диска (SSD) та жорсткого диска (HDD) в одному пристрої. Дані зберігаються на жорсткому диску, але часто використовувані дані зберігаються на твердотільному диску, що дозволяє отримати швидкий доступ до цих даних.

Змінний диск — це збірка змінних дисків, що дозволяє зберігати, переміщувати та передавати інформацію з одного комп'ютера на інший. Такі диски можуть бути використані для зберігання даних, а також для створення резервних копій.

RAM-диск — це віртуальний диск, який базується на оперативній пам'яті комп'ютера. Його можна використовувати як звичайний диск для зберігання даних, але він працює значно швидше за традиційний жорсткий диск, оскільки оперативна пам'ять набагато швидша за жорсткий диск. Зазвичай RAM-диски використовуються для тимчасового зберігання даних, що дозволяє прискорити роботу програм, які часто взаємодіють з жорстким диском. Однак, якщо комп'ютер вимикається або перезавантажується, дані на RAM-диску втрачаються, оскільки ця пам'ять є тимчасовою.

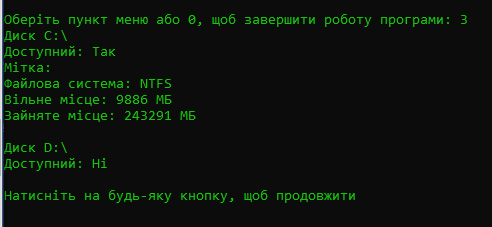


Рис. 2.3. Інформація про диски в системі та про їх файлові системи

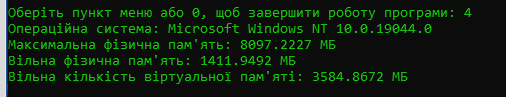


Рис. 2.4. Інформація про системну пам’ять

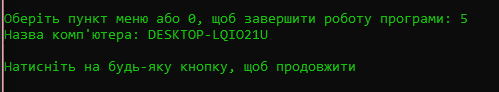


Рис. 2.5. Інформація про назву комп’ютера

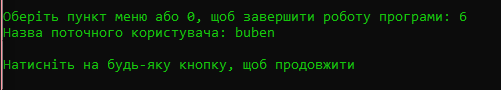


Рис. 2.6. Інформація про назву поточного користувача

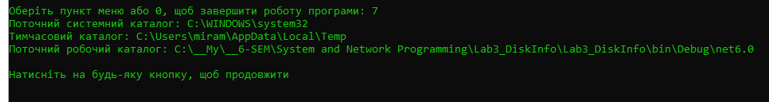


Рис. 2.7. поточний системний каталог, тимчасовий каталог, поточний робочий каталог

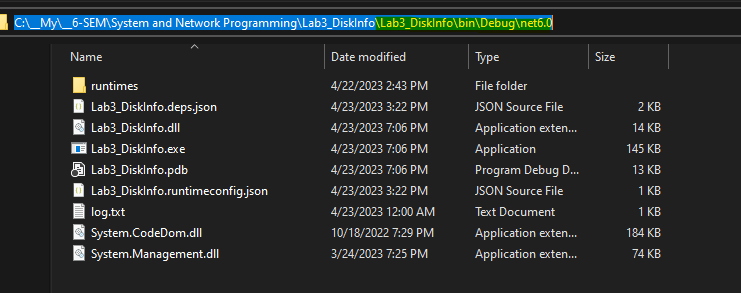


Рис. 2.8.1. Шлях до файлу логування

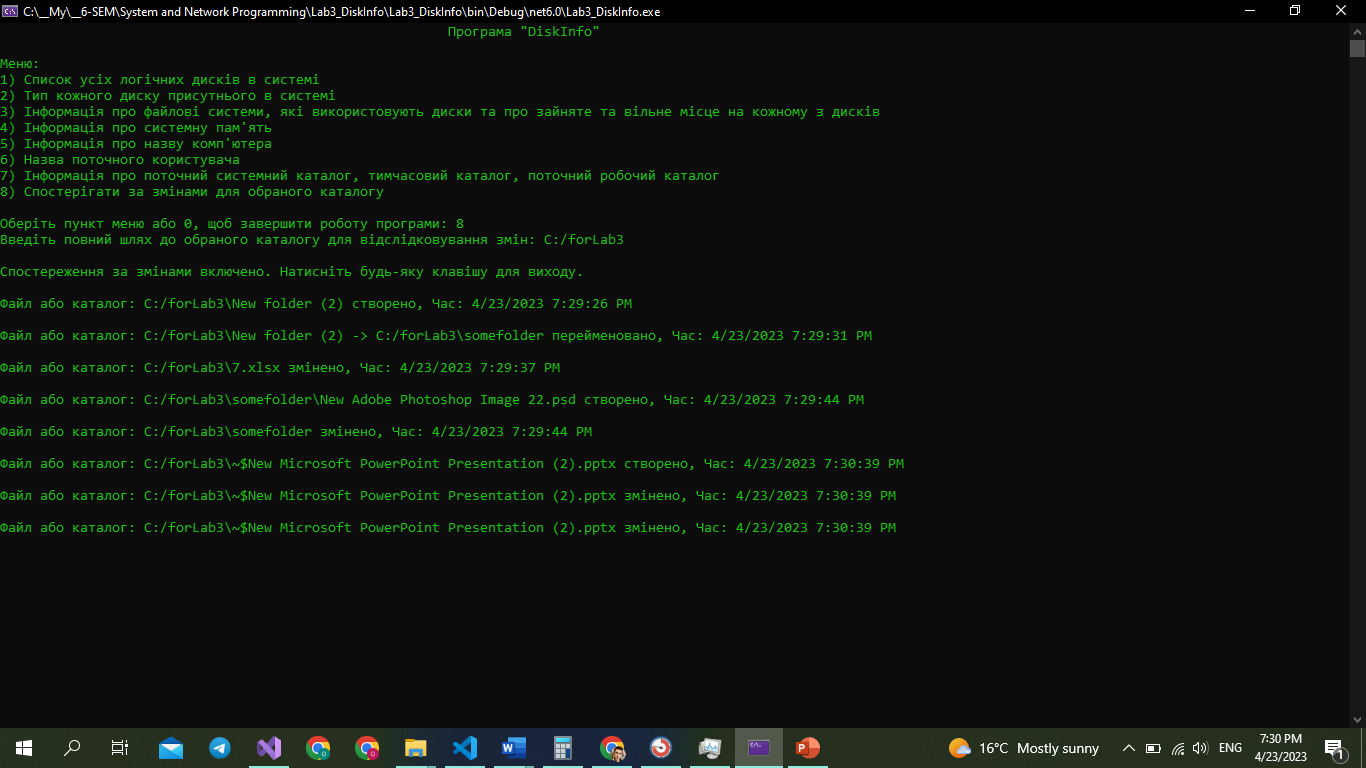


Рис. 2.8.2. Демонстрація відслідковування змін

Приклад лістингу файлу логування:

Лог-файл створено: 4/22/2023 11:30:39 PM

Файл або каталог: C:/forLab3\8\65.docx -> C:/forLab3\8\1.docx перейменовано, Час: 4/22/2023 11:59:32 PM

Файл або каталог: C:/forLab3\8 змінено, Час: 4/22/2023 11:59:32 PM

Файл або каталог: C:/forLab3\8\1.docx змінено, Час: 4/22/2023 11:59:32 PM

Файл або каталог: C:/forLab3\New folder створено, Час: 4/23/2023 7:27:48 PM

Файл або каталог: C:/forLab3\New Mathcad Document.mcd створено, Час: 4/23/2023 7:27:57 PM

Файл або каталог: C:/forLab3\6.docx змінено, Час: 4/23/2023 7:28:06 PM

Файл або каталог: C:/forLab3\~$6.docx створено, Час: 4/23/2023 7:28:06 PM

Файл або каталог: C:/forLab3\~$6.docx змінено, Час: 4/23/2023 7:28:06 PM

Файл або каталог: C:/forLab3\~WRD4013.tmp створено, Час: 4/23/2023 7:28:10 PM

Файл або каталог: C:/forLab3\~WRD4013.tmp змінено, Час: 4/23/2023 7:28:10 PM

Файл або каталог: C:/forLab3\~WRD4013.tmp змінено, Час: 4/23/2023 7:28:10 PM

Файл або каталог: C:/forLab3\~WRD4013.tmp змінено, Час: 4/23/2023 7:28:10 PM

Файл або каталог: C:/forLab3\~WRD4013.tmp змінено, Час: 4/23/2023 7:28:10 PM

Файл або каталог: C:/forLab3\~WRD4013.tmp змінено, Час: 4/23/2023 7:28:10 PM

Файл або каталог: C:/forLab3\6.docx -> C:/forLab3\~WRL4044.tmp перейменовано, Час: 4/23/2023 7:28:10 PM

Файл або каталог: C:/forLab3\~WRD4013.tmp -> C:/forLab3\6.docx перейменовано, Час: 4/23/2023 7:28:10 PM

Файл або каталог: C:/forLab3\~WRL4044.tmp змінено, Час: 4/23/2023 7:28:10 PM

Файл або каталог: C:/forLab3\6.docx змінено, Час: 4/23/2023 7:28:10 PM

Файл або каталог: C:/forLab3\6.docx змінено, Час: 4/23/2023 7:28:10 PM

Файл або каталог: C:/forLab3\~WRL4044.tmp змінено, Час: 4/23/2023 7:28:10 PM

***Посилання на реопзиторій з вихідним кодом:***

<https://gitlab.com/2020-2024/ipz-20-2/bubenko-oleksii/system_and_network_programming/lab3_diskinfo/-/tree/master/Lab3_DiskInfo>

***Висновок:*** у ході виконання лабораторної роботи я ознайомився з системними функціями, щоб отримати інформацію про дискову підсистему та використати їх на практиці