**Государственное образовательное учреждение**

**среднего профессионально образования «Тираспольский техникум информатики и права»**

**ОТЧЁТ**

**Практическое задание №3**

по дисциплине «Разработка программных модулей»

**тема: «Поиск повреждённых файлов»**

Работу выполнил

студент группы 215

Иванченко Даниил Богданович

Руководитель,

преподаватель

Подсекина Т.С.

Тирасполь, 2025

**Вариант 11**

**Задание 1**. (Directory) Найти файлы размером 0 байт или с недопустимыми символами в имени и вывести их список. (DirectoryInfo) Удалить найденные файлы и записать их список в лог.

В листинге 1 приведён код программы.

|  |
| --- |
| using System;  using System.IO;  using System.Linq;  using System.Text;  using System.Collections.Generic;  class Program  {  static void Main()  {  Console.OutputEncoding = Encoding.UTF8;  Console.Write("Введите путь к директории: ");  string targetDirectory = Console.ReadLine();  Console.Write("Введите путь для сохранения лога (например, C:\\log.txt): ");  string logFilePath = Console.ReadLine();  if (!Directory.Exists(targetDirectory))  {  Console.WriteLine("Указанная директория не существует.");  return;  }  List<string> deletedFiles = new List<string>();  DirectoryInfo dir = new DirectoryInfo(targetDirectory);  char[] invalidChars = Path.GetInvalidFileNameChars();  FileInfo[] files;  try  {  files = dir.GetFiles("\*", SearchOption.AllDirectories);  }  catch (Exception ex)  {  Console.WriteLine($"Ошибка при сканировании файлов: {ex.Message}");  return;  }  foreach (FileInfo file in files)  {  bool isZeroByte = file.Length == 0;  bool hasInvalidChars = file.Name.Any(c => invalidChars.Contains(c));  if (isZeroByte || hasInvalidChars)  {  try  {  deletedFiles.Add(file.FullName);  file.Delete();  }  catch (Exception ex)  {  Console.WriteLine($"Ошибка при удалении {file.FullName}: {ex.Message}");  }  }  }  try  {  File.WriteAllLines(logFilePath, deletedFiles, Encoding.UTF8);  Console.WriteLine("Удалённые файлы записаны в лог.");  }  catch (Exception ex)  {  Console.WriteLine($"Ошибка при записи лога: {ex.Message}");  }  Console.Read();  }  } |

Листинг 1 – программа определения алгоритмов

Ниже на рисунках 1 представлены тесты программы при разных входных параметрах.

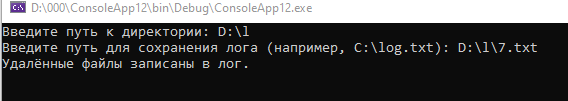
****

Рисунок 1 – тест 1