

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА
ФРАНКА

Факультет прикладної математики та інформатики

Кафедра програмування

Лабораторна робота № 2

Атрибути та властивості файлів

Виконала
студентка групи ПМО-41
Кравець Ольга

2024

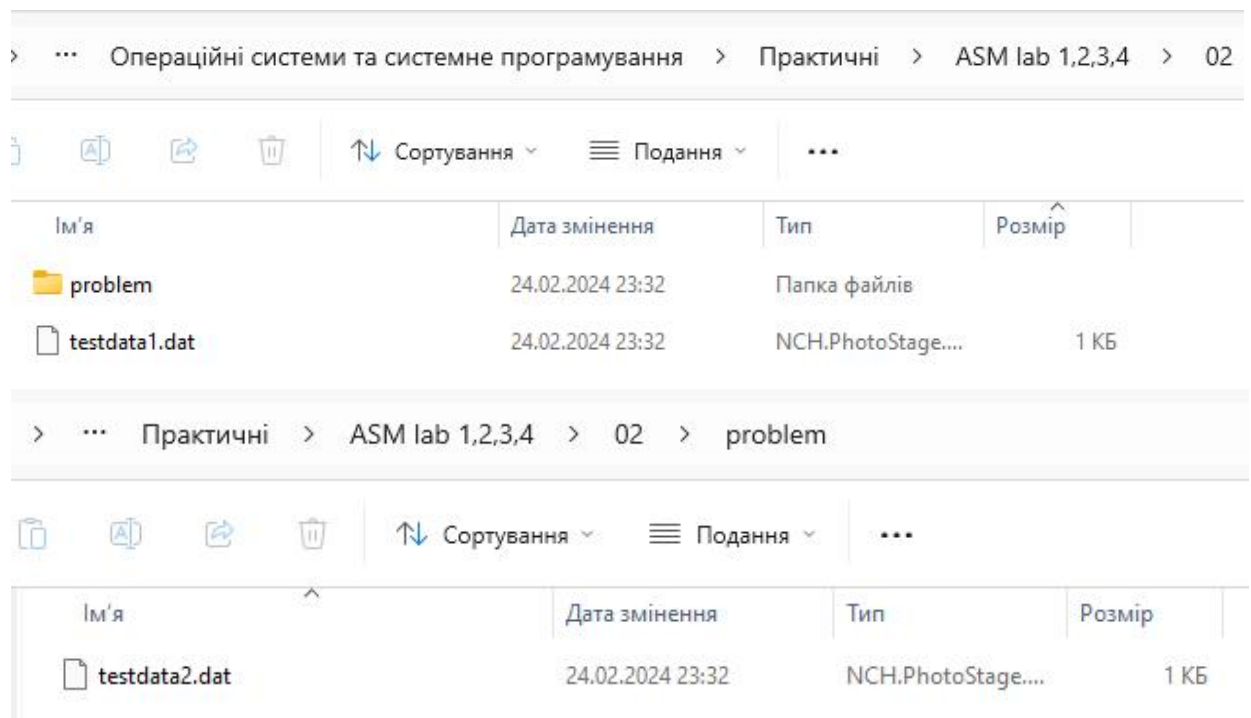
Хід роботи

Створила програму на мові Python, яка вміє шукати текстові файли з розширенням імені *.dat в поточній папці програми або у вкладеній папці *problem*, у яких записані дані як цілі або дійсні числа, необхідні для деякої задачі (у мене два файли: *testdata1.dat* та *testdata2.dat*).

Якщо окремі елементи даних є записані синтаксично неправильні з точки зору програми, то вона автоматично їх знаходить та виправляє на '0.0'. Програма у формі консольного діалогу забезпечує вибір і виконання перерахованих далі операцій з можливістю фіксування протоколу роботи в текстовому файлі *results.txt*.

Виконуються такі операції:

1) Встановлення і показ повних імен файлів *testdata1.dat* (знаходиться у поточній папці програми) та *testdata2.dat* (знаходиться у папці *problem*).



Код програми:

```
FileAttributeProperty.py X
FileAttributeProperty.py > ...
1  import glob
2  import os
3  from datetime import datetime
4  from tabulate import tabulate
5
6  # Відкриття файлу для запису результатів
7  with open("results.txt", "w") as f:
8
9      found_dat_files = glob.glob('**/*.dat', recursive=True)
10     current_path = os.getcwd()
11
12     # Виведення списку .dat файлів
13     print("**** 1 ****\n")
14     f.write("**** 1 ****\n")
15     if found_dat_files:
16         print("Знайдено наступні .dat файли:")
17         for dat_file in found_dat_files:
18             print(dat_file)
19             f.write(dat_file + "\n")
20     else:
21         print("Немає .dat файлів у поточній або вкладений папці 'problem'.")
22
23     # Виведення кількості знайдених .dat файлів
24     print(f"Всього знайдено {len(found_dat_files)} .dat файлів.")
25     f.write(f"Всього знайдено {len(found_dat_files)} .dat файлів.\n")
26
```

Результат виконання програми:

```
**** 1 ****

Знайдено наступні .dat файли:
testdata1.dat
problem\testdata2.dat
Всього знайдено 2 .dat файлів.
```

2) Побудова і показ повного шляху від кореня логічного диску до файла.

Код програми:

```
28     # Виведення абсолютних шляхів до .dat файлів
29     print("\n**** 2 ****\n")
30     f.write("\n**** 2 ****\n")
31     if found_dat_files:
32         print("Повний шлях від кореня логічного диску до .dat файлів:")
33         for dat_file in found_dat_files:
34             absolute_path = os.path.abspath(dat_file)
35             print(absolute_path)
36             f.write(absolute_path + "\n")
37     else:
38         print("Немає .dat файлів у поточній або вкладений папці 'problem'.")
39
```

Результат виконання програми:

**** 2 ****

Повний шлях від кореня логічного диску до .dat файлів:

D:\Навчання\8 семестр\Операційні системи та системне програмування\Практичні\ASM lab 1,2,3,4\02\testdata1.dat

D:\Навчання\8 семестр\Операційні системи та системне програмування\Практичні\ASM lab 1,2,3,4\02\problem\testdata2.dat

3) Отримання і показ в формі таблиці всіх властивостей і атрибутів файла, які можна отримати: повне ім'я, розмір, дату створення та редагування.

Код програми:

(форматувала дані в таблицю за допомогою бібліотеки tabulate)

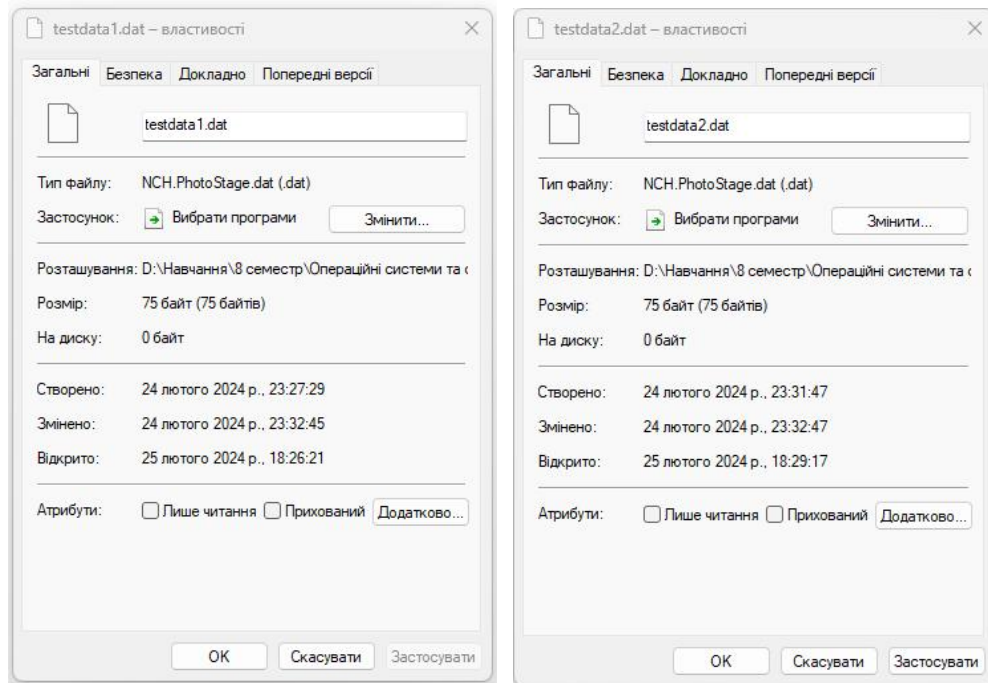
```
41 # Функція для отримання інформації про .dat файл
42 def get_file_info(file_path):
43     file_attributes = os.stat(file_path)
44     file_name = os.path.basename(file_path)
45     file_size = file_attributes.st_size
46     file_creation_time = os.path.getctime(file_path)
47     file_modification_time = os.path.getmtime(file_path)
48     file_access_time = os.path.getatime(file_path)
49
50     creation_time = datetime.fromtimestamp(file_creation_time).strftime('%Y-%m-%d %H:%M:%S')
51     modification_time = datetime.fromtimestamp(file_modification_time).strftime('%Y-%m-%d %H:%M:%S')
52     access_time = datetime.fromtimestamp(file_access_time).strftime('%Y-%m-%d %H:%M:%S')
53
54     return [file_name, file_size, creation_time, modification_time, access_time]
55
56 # Функція для форматування даних в таблицю
57 def format_table(header, data):
58     table = tabulate(data, headers=header, tablefmt="grid")
59     return table
60
61 # Отримання інформації про файли та форматування її в таблицю
62 print("\n**** 3 ****\n")
63 header = ["Назва файлу", "Розмір", "Дата створення", "Дата зміни", "Дата відкриття"]
64 data = [get_file_info(dat_file) for dat_file in found_dat_files]
65 table_result = format_table(header, data)
66
67 # Виведення таблиці у консоль
68 print(table_result)
69
70 # Запис таблиці у файл
71 f.write("\n**** 3 ****\n")
72 f.write(table_result)
```

Результат виконання програми:

**** 3 ****

Назва файлу	Розмір	Дата створення	Дата зміни	Дата відкриття
testdata1.dat	75	2024-02-24 23:27:29	2024-02-24 23:32:45	2024-02-25 20:18:46
testdata2.dat	75	2024-02-24 23:31:47	2024-02-24 23:32:47	2024-02-25 20:24:52

Отримала програмно дані, аналогічні до тих, які можна побачити в Провіднику Windows, коли натиснути правою клавішею мишки на імені файла і обрати з спадного меню пункт Властивості. Приклад скріншота цих даних:



4) Показ тексту файлу *testdata1.dat* у консолі. Програма у формі діалогу запитує який файл відкрити з тих, що є доступні.

Код програми:

```
75 # Виведення списку файлів та читання вмісту обраного файлу
76 print('\n**** 4 ****')
77 f.write('\n**** 4 ****\n')
78 if found_dat_files:
79     print("Доступні файли:")
80     f.write("\nДоступні файли:\n")
81     for i, dat_file in enumerate(found_dat_files, start=1):
82         print(f"{i}. {dat_file}")
83         f.write(f"{i}. {dat_file}\n")
84
85     file_index = input("Введіть номер файлу для перегляду: ")
86     try:
87         selected_file = found_dat_files[int(file_index) - 1]
88         with open(selected_file, "r", encoding="utf-8") as file:
89             text = file.read()
90             print(text)
91             f.write(text + "\n")
92     except (ValueError, IndexError):
93         print("Некоректний вибір файла.")
94         f.write("Некоректний вибір файла.\n")
95 else:
96     print("Немає файлів для відображення.")
97     f.write("Немає файлів для відображення.\n")
```


Результат виконання програми:

```
**** 4 ****
Доступні файли:
1. testdata1.dat
2. problem\testdata2.dat
Введіть номер файлу для перегляду: 1
1 2 3 4. 5 6is 7 8 9e s10 11@ 12 13 14h 15 16 17 18q 19 20 21 s22 23 .24 25
```

5) Рахування суми чисел файла *testdata1.dat*, виправивши неправильні записи даних числом 0.0 та повідомлення скільки виправлень відбулося.

Код програми:

```
100 # Розрахунок суми чисел у файлі, з можливістю корекції некоректних записів
101 print('\n**** 5 ****')
102 f.write("\n**** 5 ****\n")
103 filename = input("Введіть назву файлу для підрахунку суми чисел: ")
104 try:
105     with open(filename, "r", encoding="utf-8") as file:
106         text = file.read()
107         numbers = [float(num) if num.replace(".", "", 1).isdigit() else 0.0 for num in text.split()]
108         total = sum(numbers)
109         if 0.0 in numbers:
110             print("Знайдені неправильні записи, виправлено на '0.0'")
111             f.write("Знайдені неправильні записи, виправлено на '0.0'\n")
112             print(f"Всього виправлено {numbers.count(0.0)} неправильних записів.")
113             f.write(f"Всього виправлено {numbers.count(0.0)} неправильних записів.\n")
114         else:
115             correction = input("Введіть значення для виправлення неправильних записів: ")
116             if correction.replace(".", "", 1).isdigit():
117                 correction_value = float(correction)
118                 numbers = [correction_value if num == 0.0 else num for num in numbers]
119                 total = sum(numbers)
120                 print(f"Знайдені неправильні записи, виправлено на '{correction}'")
121                 f.write(f"Знайдені неправильні записи, виправлено на '{correction}'\n")
122                 print(f"Всього виправлено {numbers.count(0.0)} неправильних записів.")
123                 f.write(f"Всього виправлено {numbers.count(0.0)} неправильних записів.\n")
124             else:
125                 print("Виправлення не здійснено через неправильне значення.")
126                 f.write("Виправлення не здійснено через неправильне значення.\n")
127             print("Загальна сума чисел:", total)
128             f.write("Загальна сума чисел: " + str(total) + "\n")
129 except FileNotFoundError:
130     print(f"Файл '{filename}' не знайдено.")
131     f.write(f"Файл '{filename}' не знайдено.\n")
```

Результат виконання програми:

```
**** 5 ****
Введіть назву файлу для підрахунку суми чисел: testdata1.dat
Знайдені неправильні записи, виправлено на '0.0'
Всього виправлено 7 неправильних записів.
Загальна сума чисел: 211.24
```

Щоб користувач завершив роботу програми, має натиснути *Enter*.

```
133 # Чек на введення користувача перед завершенням програми
134 input("Натисніть Enter для завершення програми...")
```

Протокол виконання роботи програми, записаний у *results.txt*:

```
results
Файл  Редагувати  Переглянути

**** 1 ****
testdata1.dat
problem\testdata2.dat
Всього знайдено 2 .dat файлів.

**** 2 ****
D:\Навчання\8 семестр\Операційні системи та системне програмування\Практичні\ASM lab 1,2,3,4\02\testdata1.dat
D:\Навчання\8 семестр\Операційні системи та системне програмування\Практичні\ASM lab 1,2,3,4\02\problem\testdata2.dat

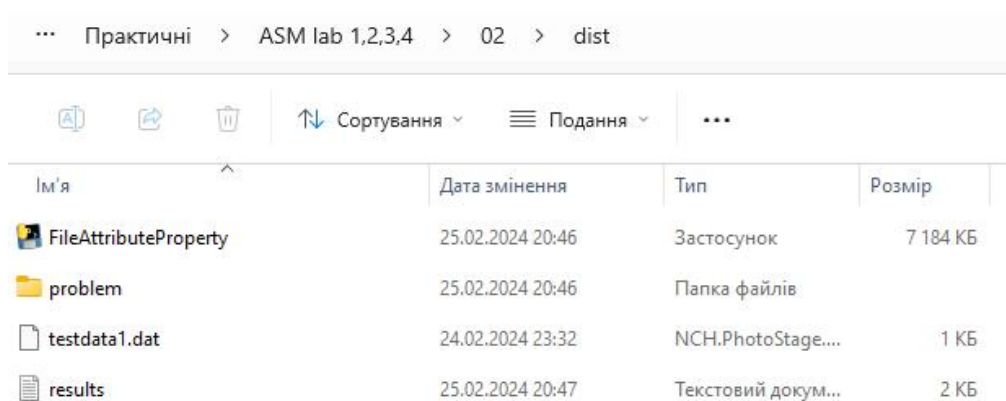
**** 3 ****
+-----+-----+-----+-----+-----+
| Назва файлу | Розмір | Дата створення | Дата зміни | Дата відкриття |
+-----+-----+-----+-----+-----+
| testdata1.dat | 75 | 2024-02-24 23:27:29 | 2024-02-24 23:32:45 | 2024-02-25 20:27:02 |
+-----+-----+-----+-----+-----+
| testdata2.dat | 75 | 2024-02-24 23:31:47 | 2024-02-24 23:32:47 | 2024-02-25 20:24:52 |
+-----+-----+-----+-----+-----+

**** 4 ****

Доступні файли:
1. testdata1.dat
2. problem\testdata2.dat
1 2 3 4. 5 6is 7 8 9e s10 11@ 12 13 14h 15 16 17 18q 19 20 21 s22 23 .24 25

**** 5 ****
Знайдені неправильні записи, виправлено на '0.0'
Всього виправлено 7 неправильних записів.
Загальна сума чисел: 211.24
```

Програма може запускатися на виконання безпосередньо без середовища програмування. Для цього був використаний інструмент збірки і пакування *pyinstaller*. Щоб програма коректно працювала достатньо щоб в одній папці був застосунок і файли, які будуть виконуватися.



Так як це консольна програма, то відкривається термінал, де можна виконати необхідні дії.

```
D:\Навчання\8 семестр\Операційні системи та системне програмування\Практичні\ASM lab 1,2,3,4\02\dist\testdata1.dat
D:\Навчання\8 семестр\Операційні системи та системне програмування\Практичні\ASM lab 1,2,3,4\02\dist\problem\testdata2.dat

**** 2 ****

Повний шлях від кореня логічного диску до .dat файлів:
D:\Навчання\8 семестр\Операційні системи та системне програмування\Практичні\ASM lab 1,2,3,4\02\dist\testdata1.dat
D:\Навчання\8 семестр\Операційні системи та системне програмування\Практичні\ASM lab 1,2,3,4\02\dist\problem\testdata2.dat

**** 3 ****

+-----+-----+-----+-----+-----+
| Назва файлу | Розмір | Дата створення | Дата зміни | Дата відкриття |
+-----+-----+-----+-----+-----+
| testdata1.dat | 75 | 2024-02-25 20:46:44 | 2024-02-24 23:32:45 | 2024-02-25 20:46:44 |
+-----+-----+-----+-----+-----+
| testdata2.dat | 75 | 2024-02-25 20:46:44 | 2024-02-24 23:32:47 | 2024-02-25 20:46:44 |
+-----+-----+-----+-----+-----+

**** 4 ****

Доступні файли:
1. testdata1.dat
2. problem\testdata2.dat
Введіть номер файлу для перегляду: 1
1 2 3 4. 5 6is 7 8 9e s10 11@ 12 13 14h 15 16 17 18q 19 20 21 s22 23 .24 25

**** 5 ****

Введіть назву файлу для підрахунку суми чисел: testdata1.dat
Знайдені неправильні записи, виправлено на '0.0'
Всього виправлено 7 неправильних записів.
Загальна сума чисел: 211.24
Натисніть Enter для завершення програми...

Вибрано елементів: 1 /,01 Mb
```