

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Кафедра «Електронних обчислювальних машин»



Звіт
з лабораторної роботи № 8
з дисципліни: «Кросплатформенні засоби програмування»
на тему: «Дослідження базових конструкцій мови Python»

Виконав:

студент групи КІ-306

Гапонова Дарина

Прийняв:

доцент кафедри ЕОМ

Іванов Ю. С.

Львів – 2023

Мета роботи: оволодіти навиками використання засобів мови Python для роботи з файлами.

Завдання (варіант № 4)

4. $y = \cos(x)/\sin(x)$

1. Написати та налагодити програму на мові Python згідно варіанту. Програма має

задовольняти наступним вимогам:

- програма має розміщуватися в окремому модулі;
- програма має реалізувати функції читання/запису файлів у текстовому і двійковому форматах результатами обчислення виразів згідно варіанту;
- програма має містити коментарі.

2. Завантажити код на GitHub згідно методичних вказівок по роботі з GitHub.

3. Скласти звіт про виконану роботу з приведенням тексту програми, результату її

виконання та фрагменту згенерованої документації та завантажити його у ВНС.

4. Дати відповідь на контрольні запитання.

Вихідний код програми

Файл lab8.py

```
import os
import struct
import math
import sys

def writeResTxt(fName, result):
    with open(fName, 'w') as f:
        f.write(str(result))

def readResTxt(fName):
    result = 0.0
    try:
        if os.path.exists(fName):
            with open(fName, 'r') as f:
                result = f.read()
        else:
            raise FileNotFoundError(f"File {fName} not found.")
    except FileNotFoundError as e:
        print(e)
    return result

def writeResBin(fName, result):
    with open(fName, 'wb') as f:
```

```

    # See https://docs.python.org/3/library/struct.html
    f.write(struct.pack('f', result))

def readResBin(fName):
    result = 0.0
    try:
        if os.path.exists(fName):
            with open(fName, 'rb') as f:
                # See https://docs.python.org/3/library/struct.html
                result = struct.unpack('f', f.read())[0]
        else:
            raise FileNotFoundError(f"File {fName} not found.")
    except FileNotFoundError as e:
        print(e)
    return result

def calculate(x):
    return math.cos(x)/math.sin(x)

if __name__ == "__main__":
    data = float(input("Enter data: "))
    result = calculate(data)
    print(f"Result in python is : {result}")
    try:
        writeResTxt("textRes.txt", result)
        writeResBin("binRes.bin", result)
        print("Result in txt is: {0}".format(readResBin("binRes.bin")))
        print("Result in bin is: {0}".format(readResTxt("textRes.txt")))
    except FileNotFoundError as e:
        print(e)
        sys.exit(1)

```

Результат виконання програми

```

● Enter data: 3
Result in python is : -7.015252551434534
Result in txt is: -7.015252590179443
Result in bin is: -7.015252551434534

```

Відповіді на контрольні запитання

1. За допомогою якої конструкції у мові Python обробляються виключні ситуації?
- "try-except".
2. Особливості роботи блоку ехсерт?
- Блок "ехсерт" використовується для обробки виключних ситуацій, які можуть виникнути у блоку "try". Він містить код, який виконується у випадку виникнення виключної ситуації.
3. Яка функція використовується для відкривання файлів у Python?
- open()

4. Особливості використання функції `open`?
 - Функція `"open()"` дозволяє відкривати файли з різними параметрами, такими як режим відкриття, кодування і т.д.
5. В яких режимах можна відкрити файл?
 - у режимах читання ("`r`"), запису ("`w`"), додавання ("`a`") і бінарного режиму ("`b`"), а також їх комбінаціях.
6. Як здійснити читання і запис файлу?
 - Для читання файлу використовують функцію `"read()"`, а для запису - `"write()"`.
7. Особливості функцій у мові Python?
 - Функції у Python можуть мати параметри, повертати значення, і багато інших особливостей, що дозволяють створювати різноманітні функції для розв'язання завдань.
8. Для чого призначений оператор `with`?
 - Оператор `"with"` використовується для автоматичного відкриття і закриття файлів (або інших ресурсів) та забезпечення правильного їхнього використання.
9. Які вимоги ставляться до об'єктів, що передаються під контроль оператору `with`?
 - Об'єкти, що передаються під контроль оператору `"with"`, повинні мати методи `"enter"` і `"exit"`, які виконуються перед входом і виходом з контексту.
10. Як поєднуються обробка виключних ситуацій і оператор `with`?
 - Обробка виключних ситуацій і оператор `"with"` можуть комбінуватися, дозволяючи обробляти виключення у контексті `"with"` і гарантуючи закриття ресурсів навіть у випадку виникнення виключної ситуації.

Висновок

У ході виконання даної лабораторної роботи, я успішно оволоділа необхідними навичками. Основні засоби мови Python для роботи з файлами були детально вивчені, включаючи відкриття, читання, запис та закриття файлів.