Міністерство освіти і науки України

Національний університет «Львівська політехніка»



Лабораторна робота №4

з навчальної дисципліни

«Спеціалізовані мови програмування»

на тему

«Розробка ASCII ART генератора для візуалізації 2D-фігур»

*Інститут комп’ютерних наук*

*та інформаційних технологій*

*Кафедри інформаційних*

*систем та мереж*

***Виконав:***

*Студент групи РІ-31*

*Олександр ЛУЧКЕВИЧ*

***Прийняла:***

*викладач*

*Cергій ЩЕРБАК*

*Львів – 2024*

**Мета роботи:** створення Генератора ASCII-арту без використання зовнішніх бібліотек

**План роботи:**

*Завдання 1:* Введення користувача

Створіть програму Python, яка отримує введення користувача щодо слова або фрази, яку вони хочуть перетворити в ASCII-арт.

*Завдання 2:* Набір символів

Визначте набір символів (наприклад, '@', '#', '\*', тощо), які будуть використовуватися для створення ASCII-арту. Ці символи будуть відображати різні відтінки.

*Завдання 3:* Розміри Art-у

Запитайте у користувача розміри (ширина і висота) ASCII-арту, який вони хочуть створити. Переконайтеся, що розміри в межах керованого діапазону

*Завдання 4:* Функція генерації Art-у

Напишіть функцію, яка генерує ASCII-арт на основі введення користувача, набору символів та розмірів. Використовуйте введення користувача, щоб визначити, які символи використовувати для кожної позиції в Art-у.

*Завдання 5:* Вирівнювання тексту

Реалізуйте опції вирівнювання тексту (ліво, центр, право), щоб користувачі могли вибирати, як їх ASCII-арт розміщується на екрані.

*Завдання 6:* Відображення мистецтва

Відобразіть створений ASCII-арт на екрані за допомогою стандартних функцій друку Python.

*Завдання 7:* Збереження у файл

Додайте можливість зберігати створений ASCII-арт у текстовий файл, щоб користувачі могли легко завантажувати та обмінюватися своїми творіннями.

*Завдання 8:* Варіанти кольорів

Дозвольте користувачам вибирати опції кольорів (чорно-білий, відтінки сірого) для свого ASCII-арту.

*Завдання 9:* Функція попереднього перегляду

Реалізуйте функцію попереднього перегляду, яка показує користувачам попередній перегляд їх ASCII-арту перед остаточним збереженням

*Завдання 10:* Інтерфейс, зрозумілий для користувача

Створіть інтерфейс для користувача у командному рядку, щоб зробити програму легкою та інтуїтивно зрозумілою для використання.

**Код програми:**

Файл lab\_calculator/main:

import shutil

from Chars import ascii\_table

def generate\_ascii\_art(text, width, height, align, color):

ascii\_art\_lines = []

for char in text.upper():

try:

ascii\_art\_lines.append(ascii\_table[char])

except KeyError:

ascii\_art\_lines.append([' ' \* 5] \* 5)

ascii\_art = [' '.join(row) for row in zip(\*ascii\_art\_lines)]

if align == 'center':

ascii\_art = [line.center(width) for line in ascii\_art]

elif align == 'right':

ascii\_art = [line.rjust(width) for line in ascii\_art]

if len(ascii\_art) > height:

ascii\_art = ascii\_art[:height]

ascii\_art\_colored = apply\_color('\n'.join(ascii\_art), color)

return ascii\_art\_colored

def apply\_color(ascii\_art, color):

colors = {

"white": "\033[97m",

"red": "\033[91m",

"green": "\033[92m",

"yellow": "\033[93m",

"blue": "\033[94m",

"magenta": "\033[95m",

"cyan": "\033[96m"

}

reset = "\033[0m"

return f"{colors.get(color, colors['white'])}{ascii\_art}{reset}"

def save\_to\_file(ascii\_art, filename):

with open(filename, 'w') as file:

file.write(ascii\_art)

def main():

print("Welcome to ASCII Art Generator!")

text = input("Enter the text you want to convert into ASCII art: ")

alignments = ['left', 'center', 'right']

align = input(f"Choose alignment ({', '.join(alignments)}): ").lower()

if align not in alignments:

align = 'left'

colors = ["white", "red", "green", "yellow", "blue", "magenta", "cyan"]

color = input(f"Choose a color ({', '.join(colors)}): ").lower()

if color not in colors:

color = "white"

terminal\_width = shutil.get\_terminal\_size().columns

terminal\_height = shutil.get\_terminal\_size().lines

width = input(f"Enter width (or press Enter for default {terminal\_width}): ")

if not width.isdigit():

width = terminal\_width

else:

width = int(width)

height = input(f"Enter height (or press Enter for default {terminal\_height}): ")

if not height.isdigit():

height = terminal\_height

else:

height = int(height)

ascii\_art = generate\_ascii\_art(text, width, height, align, color)

print("\nYour ASCII Art:\n")

print(ascii\_art)

save\_option = input("\nDo you want to save the ASCII art to a file? (yes/no): ").lower()

if save\_option == 'yes':

filename = input("Enter the filename: ") + ".txt"

save\_to\_file(ascii\_art, filename)

print(f"ASCII art saved to {filename}!")

main()

**Висновок.**

Виконуючи ці завдання, я створив генератор ASCII-арту з нуля, та надав можливість налаштовувати символи, розміри, вирівнювання та кольори, що дозволило глибше розібратися як створюється ASCII-арт