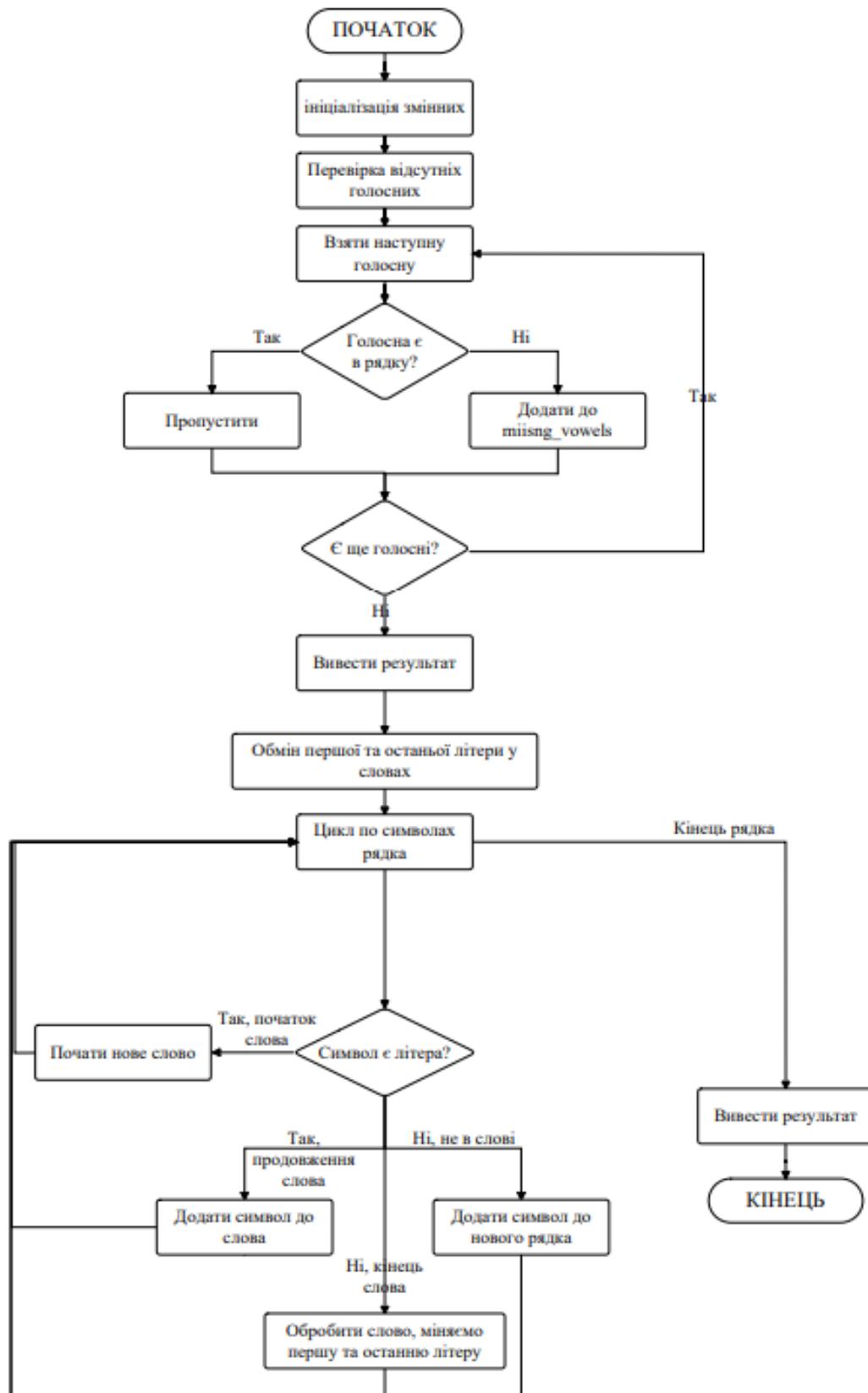


Міністерство освіти і науки України  
Донецький національний університет імені Василя Стуса  
Факультет інформаційних і прикладних технологій  
Кафедра інформаційних технологій

**З В І Т**  
з лабораторної роботи № 6  
з дисципліни «Основи програмування»  
на тему:  
«Обробка рядків у мові Python»

Виконав: студент гр. Б25\_д/F3 (Б)  
Сауляк Н. Б.  
Перевірив: доц. Бабаков Р. М.

№	Завдання
12	1) знайти голосні літери, які не містяться у рядку; 2) у кожному слові поміняти місцями першу і останню літери



Запишемо в змінну текст з яким будемо виконувати завдання:

```
S = "Вінниця – затишне місто над Південним Бугом, відоме  
своїм \n\  
яскравим фонтаном Roshen, зеленими набережними та приємною  
атмосфорою для прогулянок. \n\  
I love Vinnitsia"
```

Далі запишемо змінні з великими та маленькими усіма літерами:

```
# Голосні літери (кириличні та латинські, малі та великі)  
VOWELS = "аеїиїоуюАЕЇИЇОУЮяеіоууАЕІОУУ"  
  
# Усі літери  
UKR1 = "абвгѓдејзиіїйклмнопրстуֆхцчшшъյя"  
UKR2 = "АБВГЃДЕЄЖЗИІЇЙКЛМНОПРСТУФХЦЧШШЪЎЯ"  
ENG1 = "abcdefghijklmnopqrstuvwxyz"  
ENG2 = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"  
LETTERS = UKR1 + UKR2 + ENG1 + ENG2
```

Тепер будемо шукати відсутні голосні літери в тексті, який заданий вище:

```
# Завдання 1: Знайти голосні літери, які не містяться у рядку  
print("--- Завдання 1: Відсутні голосні літери ---")  
  
missing_vowels = "" #відсутні голосні літери  
i_vowel = 0  
len_vowels = len(VOWELS)  
  
# 1. Зовнішній цикл: перебірожної голосної літери з VOWELS  
while i_vowel < len_vowels:  
    current_vowel = VOWELS[i_vowel]  
    found_flag = 0 # 0 - не знайдено, 1 - знайдено  
    i_str = 0
```

```

len_S = len(S)

# 2. Внутрішній цикл: перебір кожного символу у вхідному
рядку S

while i_str < len_S:
    current_char = S[i_str]

    # Порівняння символів
    if current_char == current_vowel:
        found_flag = 1
        i_str = len_S  # Прискорюємо вихід з внутрішнього
цикла

    i_str = i_str + 1

# 3. Перевірка результату пошуку
if found_flag == 0:
    # Перевірка на дублікати (щоб уникнути повторів типу
'a' і 'A' якщо 'a' не знайдено)
    j = 0
    is_duplicate = 0
    while j < len(missing_vowels):
        if missing_vowels[j] == current_vowel:
            is_duplicate = 1
        j = j + 1

    if is_duplicate == 0:
        missing_vowels = missing_vowels + current_vowel +
" "

```

```

    i_vowel = i_vowel + 1

if len(missing_vowels) > 0:
    print(f"Відсутні голосні:", missing_vowels)
else:
    print("Усі голосні літери присутні у рядку.")

```

Наступним завданням будемо змінювати першу та останню літеру місцями:

```

# Завдання 2: У кожному слові поміняти місцями першу і
останню літери

print("\n--- Завдання 2: Обмін першої та останньої літер ---")

flag = 0 # 0 - набір слова не розпочатий; 1 - набір слова
          # розпочатий

W = "" # Поточне слово

S1 = "" # Новий результатуючий рядок

i = 0

L = len(S)

# Цикл по всіх символах рядка S
while i < L:
    C = S[i]

    # 1. Якщо знайдено літеру і набір слова ще не ведеться
    # (Початок слова)
    if C in LETTERS and flag == 0:
        W = C # Додаємо у слово перший символ
        flag = 1 # Ознака початку набору слова

```

```
i = i + 1
```

```
continue
```

# 2. Якщо знайдена літера і набір слова вже ведеться  
(Продовження слова)

```
elif C in LETTERS and flag == 1:
```

```
    W = W + C # Додаємо поточний символ до слова
```

```
    i = i + 1
```

```
    continue
```

# 3. Якщо знайдений нелітерний символ і набір слова не  
ведеться (Пропуск нелітерного символу)

```
elif C not in LETTERS and flag == 0:
```

```
    S1 = S1 + C # Додаємо символ до нового рядка
```

```
    i = i + 1
```

```
    continue
```

# 4. Якщо знайдений нелітерний символ і набір слова вже  
ведеться (Кінець слова)

```
elif C not in LETTERS and flag == 1:
```

```
    # --- Обробка слова W ---
```

```
    len_W = len(W)
```

```
    new_W = ""
```

```
    if len_W < 2:
```

```
        new_W = W
```

```
    else:
```

```
        first_char = W[0] #Перша літера
```

```
        last_char = W[len_W - 1] #Друга літера
```

```

# Набір середньої частини слова (використовуємо
while)

    j_mid = 1
    middle_part = ""
    while j_mid < len_W - 1:
        middle_part = middle_part + W[j_mid]
        j_mid = j_mid + 1

    # Формування нового слова
    new_W = last_char + middle_part + first_char

    # --- Додавання до нового рядка ---
    flag = 0 # Зупиняємо набір слова
    S1 = S1 + new_W # Додаємо оброблене слово до нового
 рядка

    S1 = S1 + C # Додаємо поточний символ (після слова)
    i = i + 1
    continue

else: # Ця гілка не повинна виконуватися за коректного
алгоритму
    i = i + 1

# Перевірка на слово в кінці рядка (якщо цикл закінчився, а
flag все ще 1)
if flag == 1:
    len_W = len(W)
    new_W = ""

```

```

if len_W < 2:
    new_W = W
else:
    first_char = W[0]
    last_char = W[len_W - 1]

    j_mid = 1
    middle_part = ""
    while j_mid < len_W - 1:
        middle_part = middle_part + W[j_mid]
        j_mid = j_mid + 1

    new_W = last_char + middle_part + first_char

S1 = S1 + new_W

print(f"Початковий рядок:\n ", S)
print(f"Результатуючий рядок:\n", S1)

```

Виведення програми:

--- Завдання 1: Відсутні голосні літери ---

Відсутні голосні: А Е Є И І Ї О У Ю Я и у А Е О У Ү

--- Завдання 2: Обмін першої та останньої літер ---

Початковий рядок:

Вінниця – затишне місто над Південним Бугом, відоме своїм яскравим фонтаном Roshen, зеленими набережними та приємною атмосферою для прогулянок.

I love Vinnitsia

**Результативний рядок:**

яїнницВ – еатишнз оістм дан мівденниП мугоб, еідомв мвоїс  
мскравия монтаноф nosheR, иеленимз иабережнимн ат юриємноп  
ютмосфера ялд кромуляноп.

I eovl ainnitsiV

## Висновок

У цій лабораторній роботі було навченося обробляти рядки по символіально без використання спеціальних методів і списків. Було виконано два завдання: знайдено голосні, яких немає у рядку, та переставлено першу й останню літери у кожному слові. Програма показала, як можна працювати з текстом вручну, використовуючи цикли та умовні перевірки, зберігаючи пунктуацію і пробіли.