МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

ІНСТИТУТ КОМП’ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

### *Кафедра “Системи автоматизованого проектування”*



**Звіт**

**до лабораторної роботи №8**

на тему: ВИВЧЕННЯ БІБЛІОТЕКИ ПРИКЛАДНИХ ПРОГРАМ NLTK, ДЛЯ ОПРАЦЮ­ВАННЯ ТЕКСТІВ ПРИРОДНОЮ МОВОЮ.СТРУКТУРНЕ ПРОГРАМУВАННЯ МОВОЮ PYTHON.

ОСНОВИ ПРОГРАМУВАННЯ НА МОВІ PYTHON(частина 2) ”

з дисципліни “Комп’ютерна лінгвістика”

Виконала:

ст.гр. ПРЛм-12

Щур Н. М.

Перевірив:

Старший викладач кафедри САПР

Дупак Б. П.

Львів-2015

**МЕТА РОБОТИ:** Вивчення основ програмування на мові Python. Вивчення основ

структурного програмування мовою Python. Повторення та закріплення знань

отриманих при виконанні попередніх лабораторних робіт. Покращення

загальних навичок у програмуванні.

**КОРОТКІ ТЕОРЕТИЧНІ ВІДОМОСТІ**

**Функція, як аргумент**

Аргументи функцій, які розглядалися в попередніх лабораторних

роботах, були простими об’єктами, такими як стрічка, або структурованими,

такими як список. В Python аргументом функції також може бути і інша

функція. Python підтримує ще один спосіб визначення функцій як аргументів

іншої функції , це так званий лямбда-вираз (анонімна функція).

**Функції накопичення**

Виконання функцій накопичення починається з виділення певного

об’єму пям’яті. Під час здійснення ітерацій над вхідними даними, цей об’єм

заповнюється і тільки після цього функція повертає результат (велику

структуру або узагальнений результат). Стандартний спосіб реалізувати таку

функцію це створити пустий список, в списку накопичити дані і повернути цей

список.

**Функції вищого рівня**

Python підтримує деякі функції вищого порядку, які є стандартними для

мов функціонального програмування, таких як Haskell. Розглянемо ці функції

разом з еквівалентними виразами , які використовують list comprehensions.

**Зазначені (поіменовані) аргументи**.

У випадку коли функція має багато параметрів то досить легко

заплутатись в їх порядку при виклику функції. В мові Python реалізовано

можливість явно визначати відповідність між значннями і іменами аргументів

при виклику функції. Ключі дозволяють встановити відповідність за іменами

а не за позиціями. До параметрів можна звертатися за іменами (ключами) і

присвоювати їм значення по замовчуванню. Застосовуючи такий підхід

параметри функції можуть бути записані в довільному порядку, а також і

опущені у випадку визначення їх значень по замовчуванню

**Тексти програм на мові PYTHON**

**Варіант 11**

**Завдання 2, 4, 10, 13, 15, 17**

**Завдання 2.** Створити список слів і зберегти їх в змінній sent1. Здійснити операцію присвоювання sent2 = sent1[:].Змінити один з елементів в sent1 і перевірити чи змінився sent2. Результат письмово пояснити. Sent2 — копія списку sent1, тому заміна елемента відбувається лише у першому випадку.

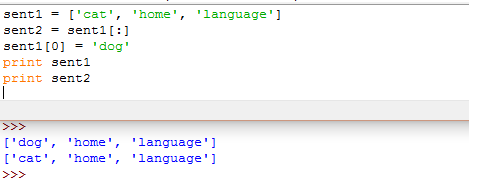


Рис. 1 . Завдання 2

**Завдання 4.** Створити список списків слів [ [‘’,’’,’’… ], [[‘’,’’,’’… ], [[‘’,’’,’’…],…] (наприклад текст складається з речень, які складаються з стрічок). Здійснити операцію присвоювання text2 = text1[:], та здійснити операцію присвоювання нового значення одному зі слів (text1[1][1] = 'Monty'). Перевірити як ці операції вплинули на text2. Результат письмово пояснити.

****

Рис. 2. Завдання 4

**Завдання 10.** Гематрія – метод виявлення прихованого змісту слів на основі

порівняння чисел, які відповідають словам. Слова з однаковими числами мають однаковий зміст. Число слова визначається сумуванням чисел, як відповідають його літерам. Здійснити аналіз корпусу (наприклад nltk.corpus.state\_union). Для кожного з текстів визначити скільки слів мають номер 555 та 777.

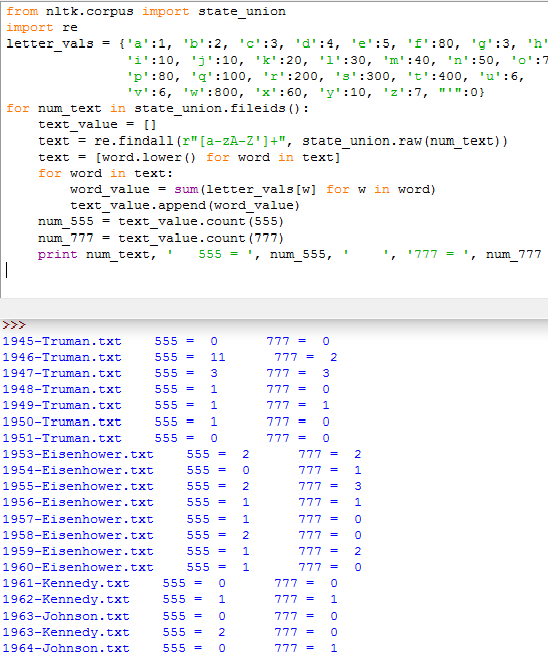


Рис. 3. Завдання 10

**Завдання 13.** Написати list comprehension для сортування списку синсетів WordNet за близькістю до заданого синсету. Наприклад, дані синсети minke\_whale.n.01,orca.n.01, novel.n.01, та tortoise.n.01, потрібно їх відсортувати згідно їх path\_distance() від right\_whale.n.01.

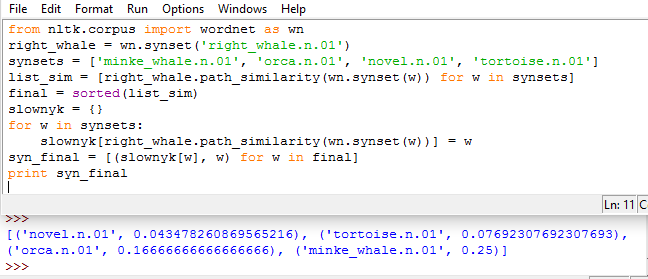


Рис. 4. Завдання 13

**Завдання 15.** Написати функцію, яка приймає текст і словник, як аргументи і повертає набір слів, які є у тексті але відсутні у словнику. Аргументи повинні бути представлені, як списки стрічок. Чи може функція мати один рядок при використанні set.difference()?

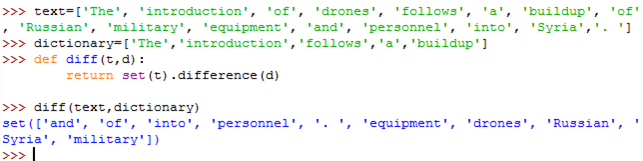


Рис. 5. Завдання 15

**Завдання 17.** В NLTK реалізовано алгоритм Левінштейна для порівняння стрічок. Спробуйте скористатись цим модулем nltk.edit\_dist(). Яким чином в цьому модулі використовується динамічне програмування? Який підхід використовується знизу-вверх чи зверху-вниз? Пояснити письмово.

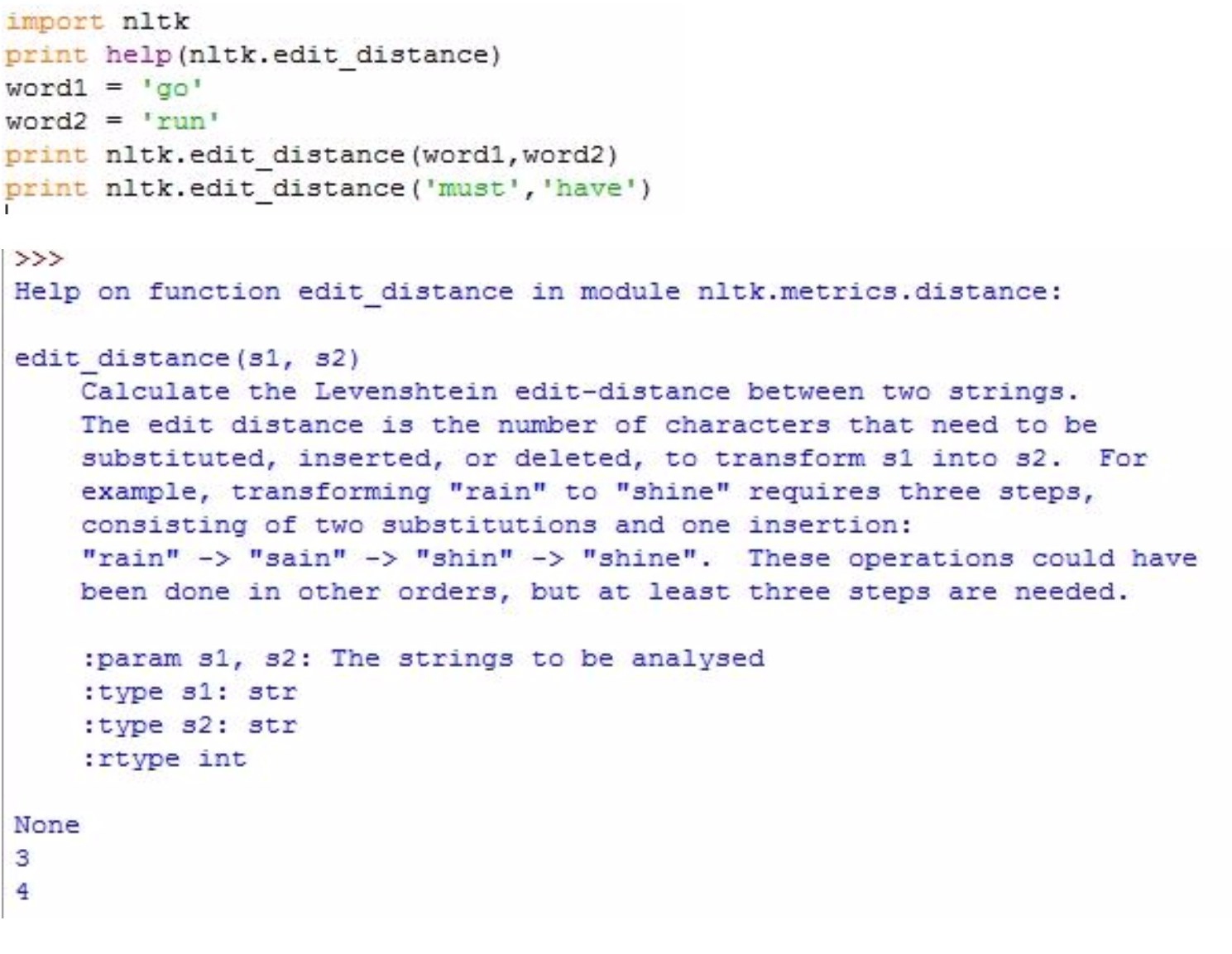


Рис. 6. Завдання 17

Відстань Левенштайна обраховує відстань між двома стрічками, використовуючи підхід знизу-вверх.

**Висновок:** на цій лабораторній роботі я ознайомилася з явищем гематрії, та навчилася обраховувати відстань Левенштайна у мові програмування Python.