МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Кафедра САПР



ЗВІТ

до лабораторної роботи № 2

на тему:

***ВИВЧЕННЯ БІБЛІОТЕКИ ПРИКЛАДНИХ ПРОГРАМ NLTK, ДЛЯ ОПРАЦЮВАННЯ ТЕКСТІВ ПРИРОДНОЮ МОВОЮ.***

***ПОЧАТКОВА ОБРОБКА ТЕКСТІВ ПРИРОДНОЮ МОВОЮ.***

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 РОБОТА**

* Вивчення основ програмування на мові *Python*.
* Вивчення методів роботи з файлами на локальних дисках та з Інтернету.
* Використання Юнікоду при обробці текстів.
* Нормалізація текстів, стемінг, лематизація та сегментація.

Корпуси текстів та тексти з Інтернету є важливими джерелами даних для здійснення лінгвістичних досліджень. Звичайно, якщо дослідник має власноруч зібрані тексти, то потрібні засоби для доступу до них.

***Доступ до текстів з Інтернету та локальних дисків***.

***Електронні книжки***

Частина електронних книжок з Project Gutenberg розповсюджується разом з NLTK у вигляді корпуса текстів. Для використання інших текстів з цього проекту можна переглянути каталог 25000 електронних книжок за адресою <http://www.gutenberg.org/catalog/> та встановити адресу (URL) потрібного текстового файлу в ASCII кодуванні. 90% текстів в Project Gutenberg є англійською мовою, але він включає також тексти більше ніж 50-ма іншими мовами (каталонська, китайська, датська, фінська, французька, німецька, італійська, португальська, іспанська…).

***Робота з HTML файлами.***

Більшість текстів в Інтернеті є у вигляді HTML документів (файлів). Інтернет сторінки можна зберігати на диску у вигляді файлів і доступатися до них. Python також дозволяє працювати Інтернет сторінками безпосередньо використовуючи функцію urlopen.

***Читання локальних файлів.***

Для читання локальних файлів необхідно використовувати вбудовану функцію Python open() та read() метод. Якщо існує файл document.txt, то змінній raw можна присвоїти його вміст:

***Ввід тексту з клавіатури.***

Для вводу тексту з клавіатури (при взаємодії користувача з програмою) потрібно використати функцію raw\_input(). Після збереження введеного тексту у змінній з ним можна працювати як зі звичайною стрічкою.

***Поняття Юнікод (Unicode).***

**Юніко́д**, ([англ.](http://uk.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D1%96%D0%B9%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE%D0%B2%D0%B0) *Unicode*) — це [промисловий стандарт](http://uk.wikipedia.org/wiki/%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D0%B8%D0%B7%D0%B0%D1%86%D1%96%D1%8F) розроблений, щоб зробити можливим для текстів і символів (графічних знаків) всіх [писемних систем](http://uk.wikipedia.org/wiki/%D0%9F%D0%B8%D1%81%D0%B5%D0%BC%D0%BD%D1%96%D1%81%D1%82%D1%8C) світу узгоджене представлення (репрезентацію) і обробку комп’ютерами. Юнікод підтримує більш ніж мільйон символів. Кожному символу ставиться у відповідність число, яке називають кодовою точкою. В Python кодові точки записуються у вигляді \u*XXXX* , де *XXXX* - чотири символи шістнадцяткового числа.

***Стемінг.***

NLTK містить декілька стандартних програм для здійснення стемінгу. Porter та Lancaster програми здійснюють стемінг (відкидання афіксів) на основі наборів правил. Porter стемер коректно обробляє слово lying (lie), а Lancaster стемер це слово обробляє з помилкою.

***Лематизація.***

WordNet лематизатор видаляє афікси тільки якщо слово, яке отримується в процесі лематизації є в його словнику. Ця процедура робить лематизацію повільнішою за стемінг.

***Сегментація***.

Токенізація це найбільш загальний випадок сегментації.

***Сегментація тексту на окремі речення.***

Робота з текстами на рівні окремих слів часто передбачає можливість поділу тексту на окремі речення. Деякі корпуси забезпечують можливість доступу на рівні окремих речень.

**Тексти програм на мові PYTHON**

**Варіант 11**

**Завдання 1,2,3,4,5,7,13,10,14**

**Завдання 1.** Напишіть функцію, яка приймає адресу URL, як аргумент, і повертає те що міститься за цією адресою з видаленням HTML розмітки. Використовувати urllib.urlopen для доступу до контенту наступним чином raw\_contents = urllib.urlopen('http://www.nltk.org/').read().



Рис. 1. Завдання1

**Завдання 2.** Збережіть деякий текст у файлі corpus.txt. Визначити функцію load(f) для читання файлу, назва якого є її аргументом і повертає стрічку, яка містить текст з файлу.

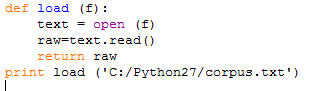
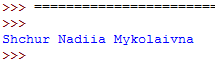
 

Рис.2. Завдання 2.

**Завдання 3**. Перепишіть наступний цикл як list comprehension:

|  |  |
| --- | --- |
|  | **>>> sent = ['The', 'dog', 'gave', 'John', 'the', 'newspaper']**  **>>> result = []**  **>>> for word in sent:**  **... word\_len = (word, len(word))**  **... result.append(word\_len)**  **>>> result**  **[('The', 3), ('dog', 3), ('gave', 4), ('John', 4), ('the', 3), ('newspaper', 9)]** |

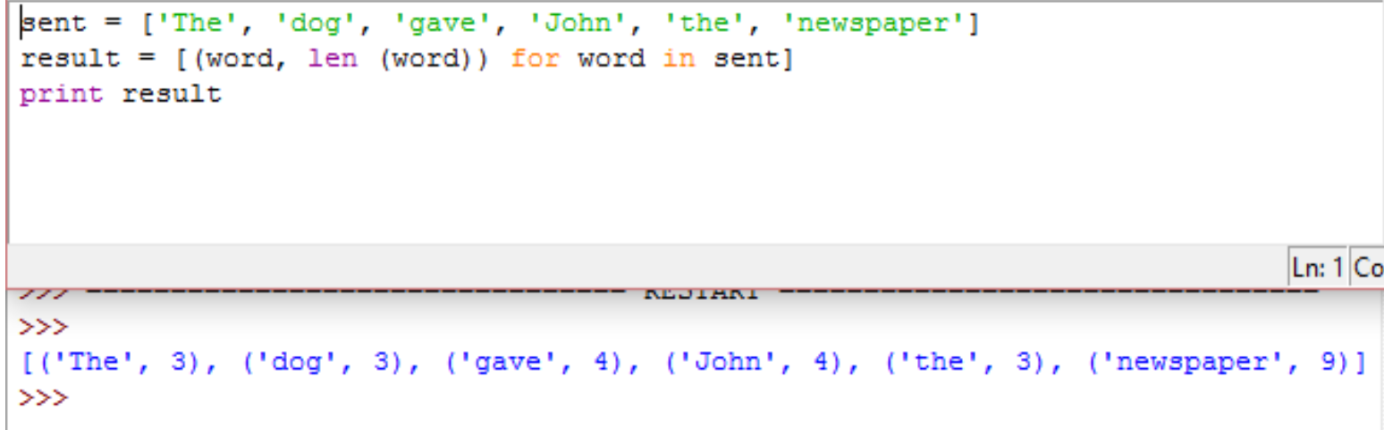


Рис. 3. Завдання 3

**Завдання 4**. Перевірити різницю між стрічками і цілим виконавши наступні дії: "3" \* 7 та 3 \* 7. Спробуйте здійснити конвертування між стрічками і цілими використавши int("3") та str(3).

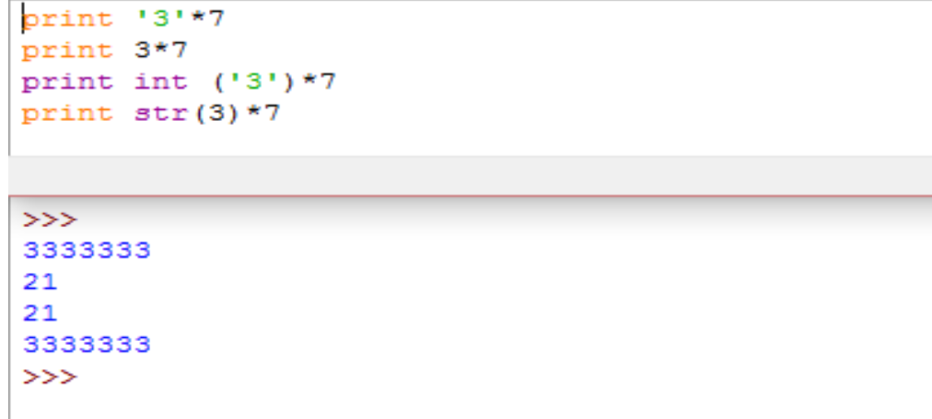


Рис. 4. Завдання 4

**Завдання 5**. Що станеться, коли стрічки форматування %6s та %-6s використовується для відображення стрічки довшої ніж 6 символів?

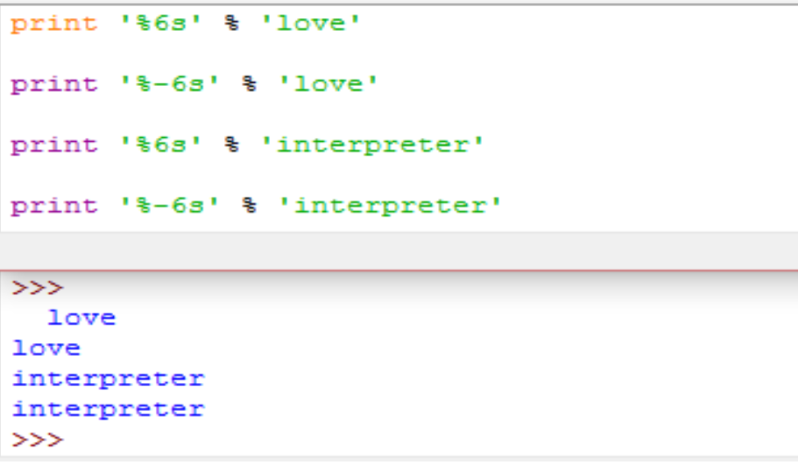


Рис. 5. Завдання 5

**Завдання 7.** Створіть файл, який буде містити слова та їх частоту записані в окремих рядках через пробіл ( fuzzy 53). Прочитайте цей файл використовуючи open(filename).readlines(). Розділіть кожну стрічку на дві частини використовуючи split(), і перетворіть число в ціле значення використовуючи int(). Результат повинен бути у вигляді списку: [['fuzzy', 53], ...].

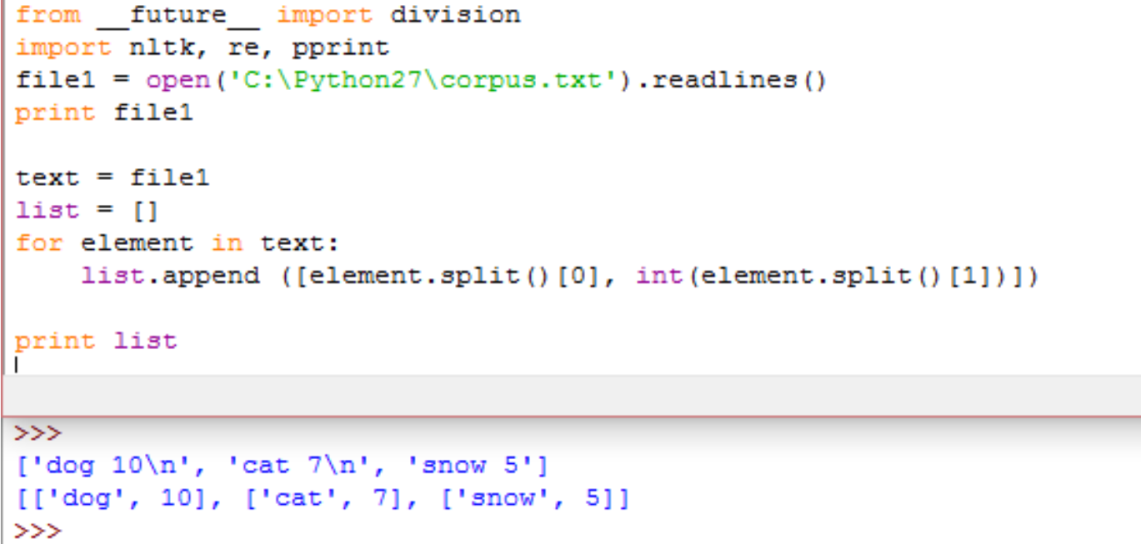


Рис.6. Завдання 7

**Завдання 13.** Використовуючи Porter стемер нормалізуйте будь-який токенізований текст . До того самого тексту застосуйте Lancaster стемер. Результати порівняйте та поясніть.

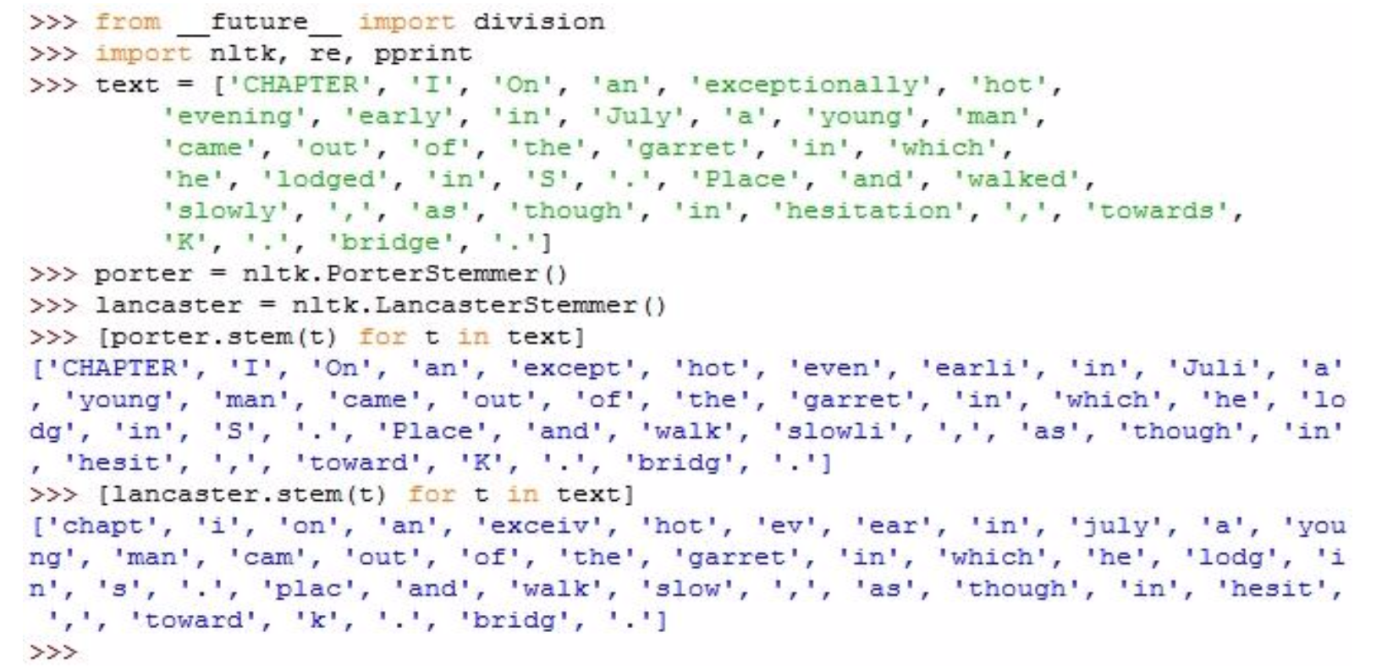


Рис.7. Завдання 13.

**Завдання 10.** Модуль random включає функцію choice(), яка випадковим чином вибирає елементи послідовності. Наприклад, choice("aehh ") буде вибирати один з чотирьох символів. Напишіть програму генерації стрічки з 500 випадково вибраних символів "aehh ". Для поєднання елементів в стрічку використовуйте ''.join() . Нормалізуйте отриманий результат використовуючи split() та join().

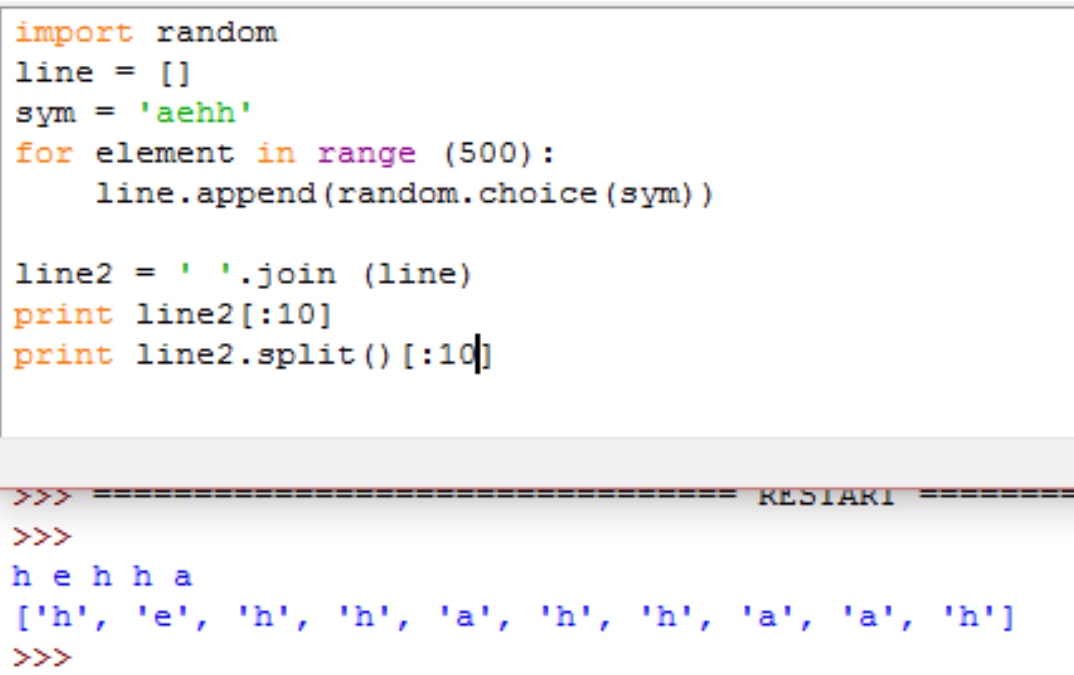


Рис. 8. Завдання 10

**Завдання 14.** Доступіться до текстів ABC Rural News та ABC Science News з корпуса (nltk.corpus.abc). Знайдіть значення для оцінки читабельності текстів (аналогічно до задачі №12). Використовуйте Punkt для поділу тексту на окремі речення.

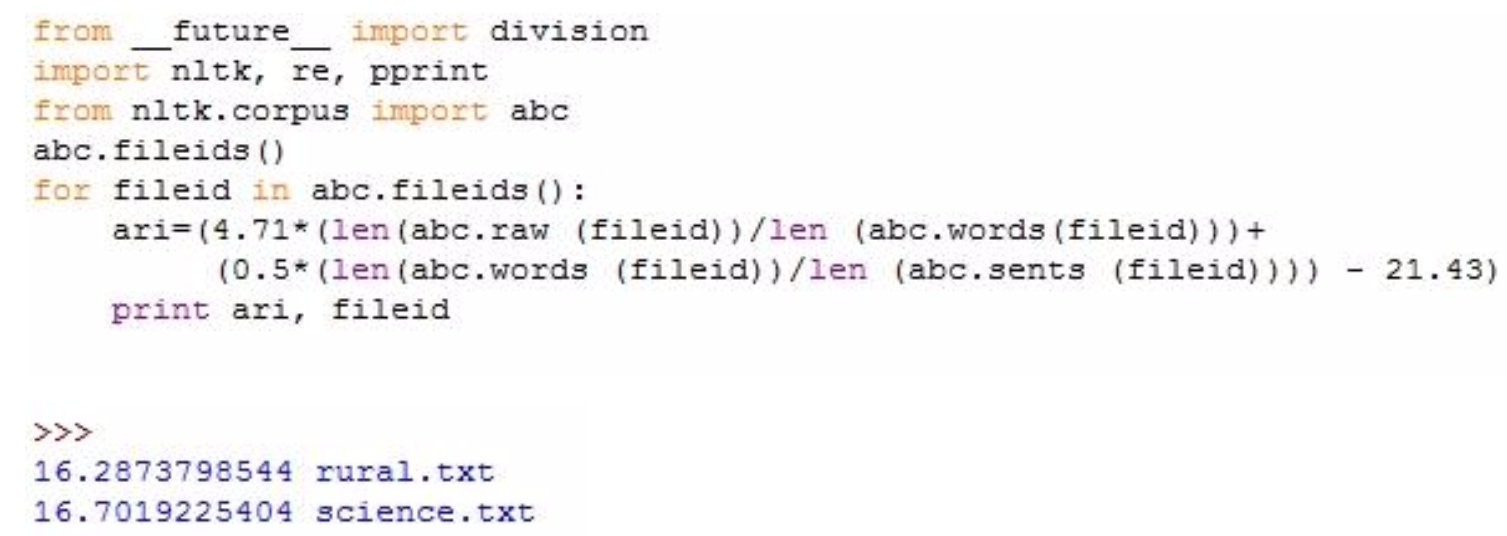


Рис. 9. Завдання 14

**Висновок:** Під час лабораторної роботи, я виконала такі завдання:

* Вивчила основи програмування на мові *Python*.
* Вивчила методи роботи з файлами на локальних дисках та з Інтернету.
* Навчилась використовувати Юнікод при обробці текстів.
* Ознайомилась із такими поняттями як нормалізація текстів, стемінг, лематизація та сегментація.