Міністерство освіти та науки України

Національний університет “Львівська політехніка”

інститут комп’ютерних наук та інформаційних технологій

Кафедра “Системи автоматизованого проектування”

Звіт

до лабораторної роботи № 3

з курсу «Комп’ютерна лінгвістика»

на тему:

«Вивчення бібліотеки прикладних програм NTLK, для опрацювання текстів природною мовою. Доступ та робота з корпусами текстів»

Виконала:

Студентка групи ПРЛм-11  
Михайлів Р.Б.

Прийняв:

Дупак Б.П.

Львів 2015

**Мета роботи**:

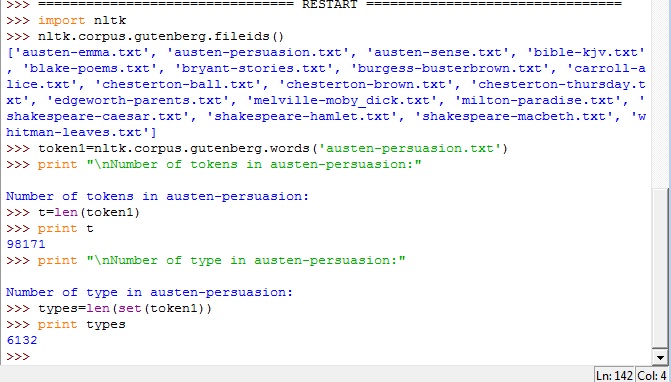
Вивчення основ програмування на мові Python. Вивчення методів доступу до корпусів текстів. Вивчення класу ConditionalFreqDist.

**Короткі теоретичні відомості**

Вирішення задач обробки текстів природною мовою передбачає використання великих об’ємів лінгвістичних даних, або інишими словами передбачає роботу з корпусами текстів. Виконання даної лабораторної роботи допоможе знайти відповідь на наступні питання: які є відомі корпуси текстів та лексичні ресурси і як отримати до них доступ використовуючи Python; які корисні конструкції має Python для виконання цієї роботи. Корпус текстів це великий набір текстів. Багато корпусів розроблені їз збереженням балансу між текстами різних жанрів, або авторів. В попередній лабораторній роботі ми працювали з промовами президентів США, які є частиною корпуса US Presidential Inaugural Addresses. З промовами ми працювали, як з одним текстом не зважаючи на те, що кожна промова має окремого автора.Обробку ми здійснювали . При роботі з копусами важливо мати засоби доступу як до окремих тексті так і до окремих частин текстів. В NLTK входить невелика частина текстів з електронного архіву текстів Project Gutenberg , який містить 25000 безкоштовних електронних книжок різних авторів (http://www.gutenberg.org/). Тексти творів в окремих файлах. Для одержання назв файлів (ідентифікаторів файлів) в яких зберігаються текстів потрібно використати наступну функцію: >>> import nltk >>> nltk.corpus.gutenberg.fileids() При програмуванні часто необхідно частину програми виконати (використати) декілька разів. Наприклад, потрібно написати програму, яка здійснює утворення множини з однини іменників і вона буде виконуватись в різних місцях програми. Швидше ніж повторювати той самий код декілька разів і більш ефективно і надійно організувати цю роботу через функцію. Функція - це програмна конструкція, яку можна викликати з одним або більше вхідними параметрами, і отримувати результат на виході. Визначаємо функцію, використовуючи ключове слово def далі потрібно дати назву функції і визначити вхідні параметри, після двокрапки записується тіло функції. Ключове слово return використовується для відображення значення, яке ми хочемо отримати на виході функції.

1. варіант

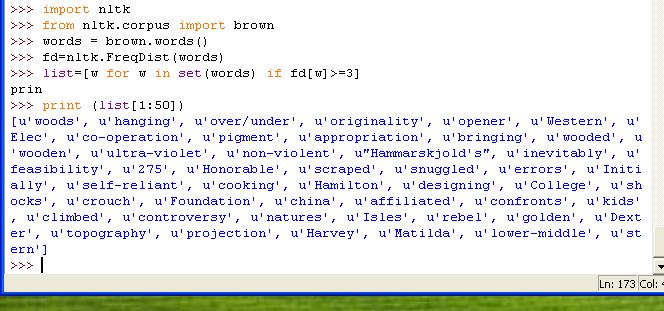
3.1 Використовуючи модуль *corpus* прочитайте текст *austin-persuasion.txt*. Визначить скільки *tokens* (слів) і *type* (унікальних слів)містить ця книжка.



3.5 Виберіть пару текстів і дослідіть відмінності між ними (кількість оригінальних слів, багатство мови, жанр). Знайдіть слова, які мають різний зміст в цих текстах, подібно до слова monstrous в *Moby Dick* та у *Sense and Sensibility*

**

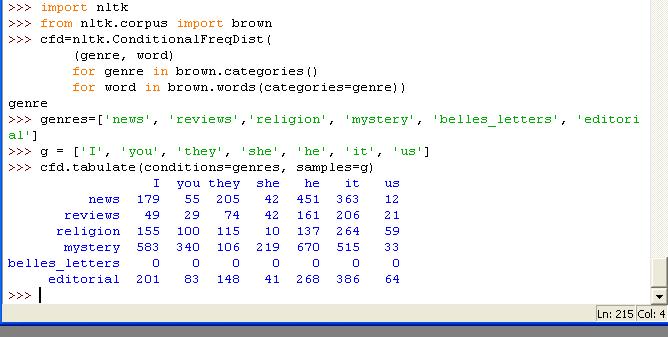
* 1. Напишіть програму для знаходження всіх слів в корпусі Brown, які зустрічаються не менш ніж три рази.



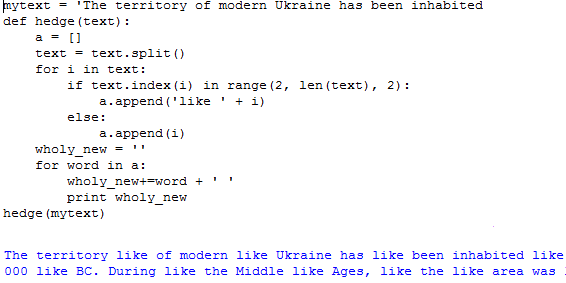
* 1. Напишіть програму генерації таблиці відношень кількість слів/кількість оригінальних слів для всіх жанрів корпуса Brown. Проаналізуйте отримані результати та поясніть їх.



* 1. Напишіть програму для створення таблиці частот слів для різних жанрів. Знайдіть слова чия присутність або відсутність є характерною для певних жанрів (подібно до модальних дієслів).



3.13Визначити функцію hedge(text), яка обробляє текст і створює нову версію цього тексту додаючи слово ‘like’ перед кожним третім словом.



**Висновок**

У ході виконання лабораторної роботи вивчила основи програмування на мові *Python*; вивчила методів доступу до корпусів текстів; вивчила клас ConditionalFreqDist.