МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

ІНСТИТУТ КОМП’ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

### *Кафедра “Системи автоматизованого проектування”*



Звіт

до лабораторної роботи №7

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:

студентка групи ПРЛм-11

Свіляк О.Ю.

Прийняв:

викладач

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.

**Теоретичні відомості**

Присвоювання – найпростіше поняття програмування, але навіть і йому властиві певні тонкощі.  За допомогою операції присвоєння відбувається копіювання значення виразу іншій змінній, хоча насправді, значення структурованого об’єкту, наприклад такого, як список, це є посилання на цей об’єкт.

Python підтримує два способи порівняння. Оператор is перевіряє об’єкти на ідентичність. Створивши список з декількох копій одного і того самого об’єкту не складно переконатися, що елементи цього списку не тільки ідентичні, згідно ==, але і є одним і тим самим об’єктом.

Змінна, тип якої – кортеж, створюється за допомогою ком і переважно обмежується дужками. Зазвичай, кортежі можуть містити будь-яку кількість елементів та членів.  Кортежі створюються за допомогою ком (оператор - кома). Для створення кортежу, який містить єдиний елемент 'snark'  , кома ставиться безпосередньо після цього елементу 'snark',.  Пустий кортеж створюється за допомогою пустих дужок (t=()).

Послідовності різних типів можна перетворювати між собою. Наприклад, tuple(s) – перетворення послідовності будь-якого типу в кортеж, list(s) – перетворення послідовності будь-якого типу в список. Для перетворення списку стрічок в єдину стрічку потрібно використовувати функцію join(), наприклад, ':'.join(words).

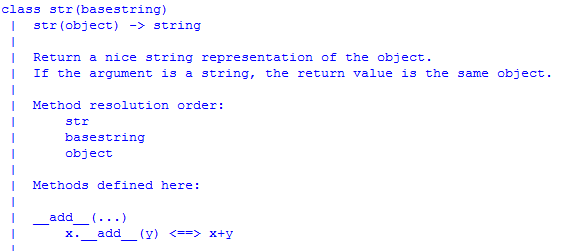
Python, за допомогою таких функцій, як sorted() and reversed(), дозволяє змінювати порядок елементів у послідовностях. Також існують функції, які модифікують структуру послідовностей, що знаходить широке використання при обробці мови. Функція, zip() приймає елементи двох або більше послідовностей і "zip" переміщує їх, попарно, один з одним, в один список пар. Маючи послідовність s, і скориставшись вбудованою функцією мови програмування Python enumerate(s) отримуємо пари, які містять індекс та елемент послідовності, який відповідає цьому індексу.

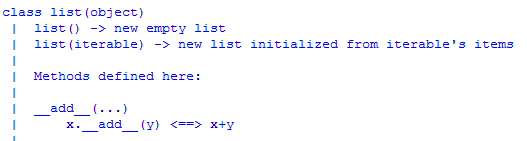
Використання list comprehensions дозволяє отримувати компактний та зручний для читання текст програми.

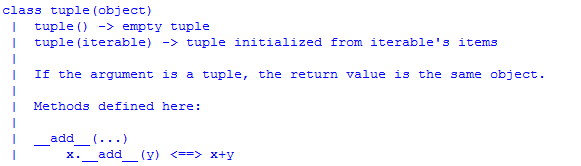
**Тексти програм на мові Python**

**Варіант 15**

1. Знайти в Python's help додаткову інформацію про послідовності. В інтерпретаторі, набрати по черзі help(str), help(list), та help(tuple). На екрані буде відображено повний список функцій властивих кожному з типів. Деякі функції мають спеціальні імена з подвійними підкреслюваннями. Кожній такій функції відповідає і інший запис показаний в документації. Наприклад x.\_\_getitem\_\_(y) відповідає x[y].







1. Знайти три операції, які можна здійснювати і зі списками та із кортежами. Знайти три операції, які не можна здійснювати над кортежами. Знайдіть коли використання списку замість кортежу приводить до Python помилки.

Спільні:

|  \_\_add\_\_(...)

|  x.\_\_add\_\_(y) <==> x+y

|  \_\_contains\_\_(...)

|  x.\_\_contains\_\_(y) <==> y in x

|  \_\_eq\_\_(...)

|  x.\_\_eq\_\_(y) <==> x==y

Не можна з кортежами:

|  \_\_imul\_\_(...)

|  x.\_\_imul\_\_(y) <==> x\*=y

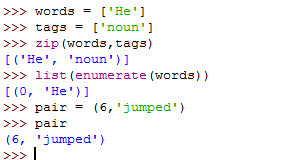
|  \_\_delitem\_\_(...)

|   x.\_\_delitem\_\_(y) <==> del x[y]

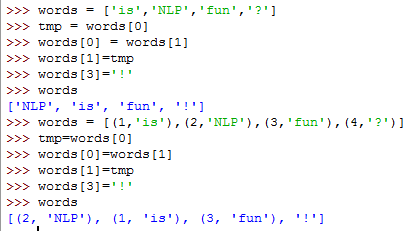
|  \_\_iadd\_\_(...)

|  x.\_\_iadd\_\_(y)<==>x+=y

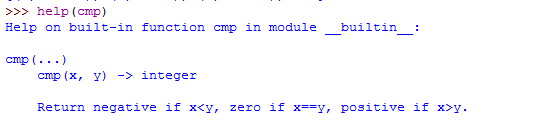
1. Яким чином можна створити кортеж з одного елемента. Продемонструвати два різні способи.

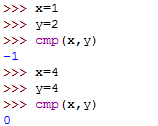


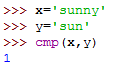
1. Створити список words = ['is', 'NLP', 'fun', '?']. Використовуючи операції присвоювання подібні до words[1] = words[2] та тимчасову змінну tmp перетворити цей список в список ['NLP', 'is', 'fun', '!']. Здійснити аналогічні перетворення використовуючи присвоювання в кортежах.



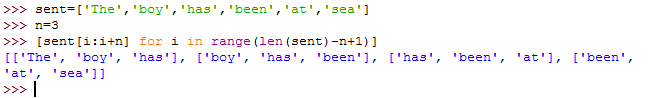
1. Прочитати про вбудовану функцію здійснення порівнянь cmp, набравши help(cmp). Продемонструвати чим поведінка цієї функції відрізняється від поведінки операторів порівняння.



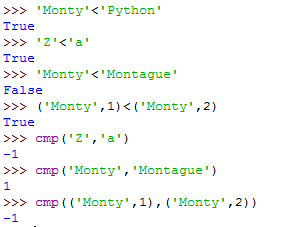




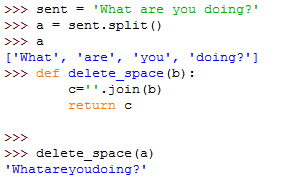
1. Написати програму для коректного виділення в тексті n-грамів з врахуванням граничних випадків: n = 1, та n = len(sent)?



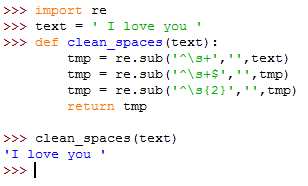
1. Використати оператори нерівності для порівняння стрічок, наприклад. 'Monty' < 'Python'. Що станеться, якщо виконати 'Z' < 'a'? Порівняти стрічки,як мають однаковий префікс, наприклад 'Monty' < 'Montague'. Спробувати порівняти структуровані об’єкти ,наприклад. ('Monty', 1) < ('Monty', 2). Чи отримали очікувані результати?



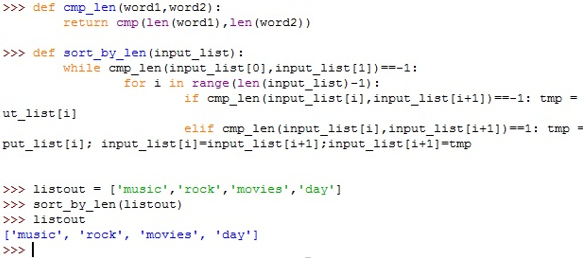
1. Написати програму видалення пробілів на початку і в кінці стрічки та для видалення зайвих пробілів між словами. Використовувати split() та join(). Оформити у вигляді функції. Функція повинна містити повну стрічку документування.



1. Написати програму видалення пробілів на початку і в кінці стрічки та для видалення зайвих пробілів між словами. Використовувати re.sub() . Оформити у вигляді функції. Функція повинна містити повну стрічку документування



1. Написати програму сортування слів за їх довжиною. Визначити допоміжну функцію cmp\_len, яка буде використовувати функцію cmp для порівняння довжин слів. Функція повинна містити повну стрічку документування.



**Висновок:** на цій лабораторній роботі я розглянула яким чином писати гарно структуровані, чіткі, сприйнятливі (читабельні) програми, придатні для неодноразового використання, яким чином працюють основні конструкційні блоки, а саме: цикл, функція, присвоювання, які можуть бути пастки при програмуванні на Python та як їх уникати.