Лабораторна робота № 4

4 години

Тема: **Методи та функції роботи з рядками. Регулярні вирази Python.**

Мета: Отримати практичні навички опрацювання текстових даних засобами методів, функцій та регулярних виразів мовою Python.

Література: http://www.matfiz.univ.kiev.ua/userfiles/files/Pres21.pdf

http://blog.dzinko.org/2011/03/python.html регулярні вирази Python.

Хід роботи.

Завдання виконати використовуючи методи роботи з рядками

1. Задано текст із 3 речень. Скласти програму, яка визначає і виводить на екран:
    a. всі його різні слова;
    b. довжину його самого короткого слова;
    c. букву а замінити на о та визначити кількість замін.
2. Напишіть програму, яка приймає від користувача рядок, і відображає цей рядок у верхньому і нижньому регістрах.
3. Дано рядок. Змініть регістр символів в цьому рядку так, щоб перша буква кожного слова була великою, а інші літери - малими.

Завдання виконати використовуючи методи роботи з регулярними виразами.

4. Задано текстову змінну з 5 речень. Використовуючи регулярні вирази визначте:
    a. кількість слів у тексті,
    b. слова, що починаються на голосну та їх кількість,
    c. слова що починаються на приголосну,
    d. з вашого тексту виберіть три будь яких слова і визначте позиції їх розміщення.
    e. з тексту виберіть слово і замініть його на ваше прізвище.
5. Задано текстову змінну з 6 слів. Отримайте список символів без пробілів та список перших двох букв кожного слова. Зі списку символів створіть новий список який буде включати всі символи крім «а», «б».
6. Створити текстовий рядок в якому використати назви мов програмування. Використовуючи регулярний вираз отримати список назв мов програмування використаних у тексті.
7. Задайте рядок зі списком електронних адрес та їх власників. Зі створеного списку виберіть домени електронних адрес.

8. Задано текст в якому є деякі послідовності цифр що повторюються (наприклад: *333334 333 123 2334).* Знайти всі числа, в яких зустрічаються послідовності цифр 5 довжиною від 2-х до 3-х символів.
9. Задано рядок в якому відображається час (год:хв:сек) та дата в повному форматі. Отримайте список часу та дат. Отримайте список годин та років.

10. У тексті зустрічаються ідентифікатори contig називаються як NODE_1, NODE_2 тощо. Вважється за краще, щоб їх називали contig1, contig2 тощо. Створити текстовий рядок з цими змінами.

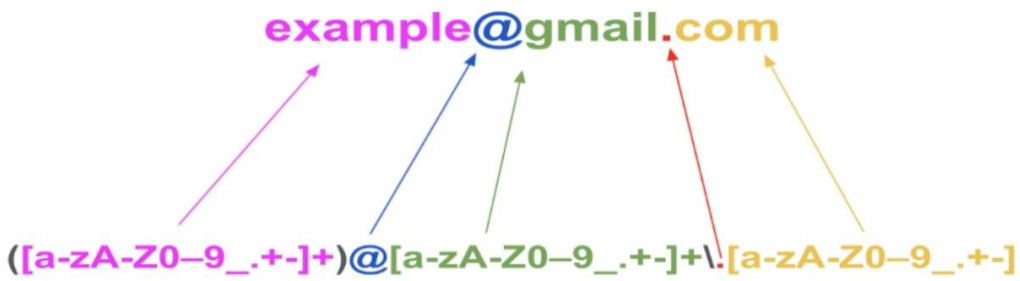
11. Розробити алгоритм програми, що перевіряє коректність введення почтового індексу засобами регулярних виразів за варіантом, що надано в табл. 1.

Табл. 1 – Індивідуальні варіанти завдання

Варіант	Країна	Алгоритм формування індексу
1	Сполучені Штати Америки	Числовий код з дев'ятьма цифрами. Після п'ятої опціонально тире
2	Нідерланди	Чотири цифри. Перша цифра не нуль. Після опціонального пропуску код з двох літер
3	Італія	П'ять цифр. Попереду опціонально "V-" чи "I-".
4	Індія	Шість цифр. Перша цифра не нуль. Опціонально пропуск після третьої
5	Польща	Дві цифри, дефіс, три цифри
6	Іспанія	П'ять цифр. Після третьої цифри опціональний пропуск. Попереду опціонально "I-".
7	Великобританія	Від 5 до 8 цифр і літер, що розділені пропуском. Наприклад, AA9A 9AA
8	Латвія	Чотири цифри Попереду опціонально "V-" чи "I-".
9	Македонія	Чотири цифри. Перша від 1 до 7
10	Португалія	Чотири цифр, дефіс, три цифри, пропуск, назва регіону до 25 літер

Методичні рекомендації.

Регулярні вирази (regular expressions) – сучасна система пошуку текстових фрагментів у електронних документах, що заснована на спеціальній системи запису зразків для пошуку.



Регулярні вирази використовують символ зворотної косої риски ('\') для позначення спеціальних форм або для дозволу використання спеціальних символів без звернення до їхнього особливого значення.

Метасимволи

[…] Набір символів

[^…] Негативний клас символів. Відповідає будь-якому символу, не укладеному у квадратні дужки

**** Повідомляє про спеціальну послідовність (також може використовуватися для екранування спеціальних символів)

. Будь-який символ (крім символу нового рядка)

^ Починається з

$ Закінчується на

***** Нуль або більше випадків

+ Один або кілька випадків

? Нуль чи одне входження

{} Рівно вказана кількість входжень

{n,m} Відповідає щонайменше «n», але з більше «m» повторень попереднього символу.

| Чергування. Відповідає символам до або після |

() Захоплення та угруповання

(xyz) Група символів. Відповідає символам xyz у цьому порядку.

Спеціальна послідовність - це коли за символом \ слідує один із символів у списку нижче, яка має особливе значення:

\A Повертає збіг, якщо вказані символи знаходяться на початку рядка

\b Повертає збіг, у якому зазначені символи знаходяться на початку або наприкінці слова

\B Повертає збіг, у якому зазначені символи присутні, але НЕ на початку (або наприкінці) слова

\d Повертає збіг, у якому рядок містить цифри (числа від 0 до 9)

\D Повертає збіг, у якому рядок НЕ містить цифр
\s Повертає збіг, у якому рядок містить символ пробілу
\S Повертає збіг, в якому рядок НЕ містить пробілу
\w Повертає збіг, у якому рядок містить будь-які символи слова (символи від a до Z, цифри від 0 до 9 та символ підкреслення _)
\W Повертає збіг, в якому рядок НЕ містить символів слова
\Z Повертає збіг, якщо вказані символи знаходяться в кінці рядка

Set (Набір) – це набір символів усередині пари квадратних дужок [] зі спеціальним значенням:

[arn] Повертає збіг, у якому є один із зазначених символів (a, r або n)
[a-n] Повертає збіг для будь-якого символу нижнього регістру в алфавітному порядку від a до n
[^arn] Повертає збіг для будь-якого символу, ЗА ВИКЛЮЧЕННЯМ a, r і n
[0123] Повертає збіг, в якому присутня будь-яка із зазначених цифр (0, 1, 2 або 3)
[0-9] Повертає збіг для будь-якої цифри від 0 до 9
[0-5][0-9] Повертає збіг для будь-яких двоцифрових чисел від 00 до 59
[a-zA-Z] Повертає відповідність для будь-якого символу в алфавітному порядку від a до z, у нижньому регістрі АБО у верхньому регістрі
[+] У наборах + *. | () $ {} знак не має особливого значення, тому [+] означає: повернути збіг для будь-якого символу + у рядку

Модуль re надає набір функцій/методів, які дозволяють нам шукати рядок зі збігом:

findall() — Повертає список, що містить усі збіги
search() — Повертає об'єкт Match, якщо десь у рядку є збіг
split() — Повертає список, в якому рядок був поділений при кожному збігу
sub() — Замінює один або кілька збігів рядком
subn() — Робить те саме, що й sub(), але повертає новий рядок і кількість замін
match() — Шукає збіг з початку рядка
finditer() - Шукає всі збіги з pattern, повертає ітератор
compile() — Компілює regular expression, на виході отримуємо об'єкт, до якого можна застосовувати всі перелічені функції
fullmatch() — Перевіряє, що весь рядок відповідає описаному регулярному виразу
flags (прапори) — Вказуються у функціях, впливають на поведінку регулярного вираження

Контрольні запитання

1. Що таке регулярний вираз?

2. Яку функціональність надає модуль re?
3. Що таке шаблони регулярних виразів та для чого вони використовуються?
4. Який синтаксис можна використовувати для побудови регулярних виразів?
5. Що працює швидше – рядкова функція чи аналогічний регулярний вираз?
6. Як створювалася теорія регулярних виразів?
7. Що вважається найпростішим регулярним виразом?
8. Що таке вираз у квадратних дужках?
9. Яке значення має дефіс у регулярному виразі?
10. Яке значення має символ ^ у регулярному виразі?
11. Яким чином реалізовані розгалуження у регулярних виразах?