**ЛАБОРАТОРНА РОБОТА №2.**

**Тема:** Функції у мові Python

**Мета:** навчитися створювати власні функції у мові Python.

**Теоретична частина**

Функція в python – об'єкт, який приймає аргументи і повертає значення. Зазвичай функція визначається за допомогою інструкції def.

Визначимо найпростішу функцію:

def add(x, y):

return x + y

Інструкція return каже, що потрібно повернути значення. У нашому випадку функція повертає суму x і y. Тепер ми її можемо викликати:

**>>>**  
**>>>**add(1,10)  
11  
**>>>**add('abc','def')  
'abcdef'

Функція може бути будь-якої складності і повертати будь-які об'єкти  
(списки, кортежі, і навіть функції). Функція може і не закінчуватися інструкцією return, при цьому функція поверне значення None:

**>>>**  
**>>>def**func():  
**...pass**  
**...**  
**>>>print**(func())  
None

**Аргументи функції**  
Функція може приймати будь-яку кількість аргументів чи не приймати їх  
зовсім. Також поширені функції з довільним числом аргументів, функції з  
позиційними і іменованими аргументами, обов'язковими і необов'язковими.  
**>>>**  
**>>> def** func(a, b, c=2): *# c - необов'язковий аргумент*  
**... return** a + b + c  
**...**  
**>>>** func(1, 2) *# a = 1, b = 2, c = 2 (за замовчуванням)*  
5  
**>>>** func(1, 2, 3) *# a = 1, b = 2, c = 3*  
6  
**>>>** func(a=1, b=3) *# a = 1, b = 3, c = 2*  
6  
**>>>** func(a=3, c=6) *# a = 3, c = 6, b не визначений - виникне помилка*  
20  
Функція також може приймати змінну кількість позиційних аргументів,  
тоді перед ім'ям ставиться \*:  
**>>>**  
**>>> def** func(\*args):  
**... return** args  
**...**  
**>>>** func(1, 2, 3, 'abc')  
(1, 2, 3, 'abc')  
**>>>** func()  
()  
**>>>** func(1)  
(1,)  
Як видно з прикладу, args – це кортеж з усіх переданих аргументів функції, і зі змінною можна працювати так само, як і з кортежем. Функція може приймати і довільне число іменованих аргументів, тоді перед ім'ям ставиться \*\*:  
**>>>**  
**>>> def** func(\*\*kwargs):  
**... return** kwargs  
**...**  
**>>>** func(a=1, b=2, c=3)  
{'a': 1, 'c': 3, 'b': 2}  
**>>>** func()  
{}  
**>>>** func(a='python')  
{'a': 'python'}  
В змінній kwargs у нас зберігається словник, з яким ми, знову-таки,  
можемо робити все, що нам заманеться.  
**Анонімні функції, інструкція lambda**  
Анонімні функції можуть містити лише один вислів, але і виконуються  
вони швидше. Анонімні функції створюються за допомогою інструкції lambda.  
Крім цього, їх не обов'язково привласнювати змінній, як ми робили з  
інструкцією def func ():  
**>>>**  
**>>>** func = **lambda** x, y: x + y  
**>>>** func(1, 2)  
3  
**>>>** func('a', 'b')  
'ab'  
**>>>** (**lambda** x, y: x + y)(1, 2)  
3  
**>>>** (**lambda** x, y: x + y)('a', 'b')  
'ab'  
lambda функції, на відміну від звичайної, не потрібна інструкція return, а в  
іншому, вона поводиться точно так же:  
**>>>**  
**>>>** func = **lambda** \*args: args  
**>>>** func(1, 2, 3, 4)  
(1, 2, 3, 4)  
f = **lambda** x: x\*\*2 + 4  
Те ж саме, що:  
def f(x):  
return x\*\*2 + 4  
У загальному випадку будь-яка конструкція виду:  
def g(arg1, arg2, arg3, ...):  
return expression  
Може бути записана як:  
g = **lambda** arg1, arg2, arg3, ...:expression  
Lambda-функції дуже зручні для того, щоб визначати невеликі функції " на льоту" і тому дуже популярні серед багатьох програмістів. Lambda-функції часто використовуються для швидкого визначення функції як аргумент іншої функції.

**Завдання 1:**

Задано дійсні величини . Розробити програму, яка обчислює значення суми з заданою точністю і вказує кількість врахованих доданків.

1.

2.

3.

4.

5.

6.

7.

8.

9.

10.

**Завдання 2:**

Задано дійсні величини . Розробити програму, яка знаходить і друкує перший член і його номер у заданій послідовності, для якого виконується умова . Обмежитись розглядом перших членів послідовності.

1.

2.

3.

4.

5.

6.

7.

8.

9.

10.

**Завдання 3:**

1. Задано текст, слова в якому розділені пробілами і розділовими знаками. Розробити програму, яка вилучає в кожному слові цього тексту всі наступні входження першої літери.

2. Задано текст, слова в якому розділені пробілами і розділовими знаками. Розробити програму, яка вилучає в кожному слові цього тексту всі попередні входження останньої літери.

3. Задано текст, слова в якому розділені пробілами і розділовими знаками. Розробити програму, яка вилучає з цього тексту всі повторні входження слів.

4. Задано текст, слова в якому розділені пробілами і розділовими знаками. Розробити програму, яка знаходить і вилучає всі слова, що входять в цей текст по одному разу.

5. Задано текст, слова в якому розділені пробілами і розділовими знаками. Розробити програму, яка в словах непарної довжини цього тексту вилучає середню літеру.

6. Задано текст, слова в якому розділені пробілами і розділовими знаками. Розробити програму, яка знаходить і друкує всі слова, що входять у заданий текст по одному разу.

7. Задано текст, слова в якому розділені пробілами і розділовими знаками, та два окремих слова. Розробити програму, яка замінює всі входження в заданий текст першого слова другим словом.

8. Задано два тексти, слова в яких розділені пробілами і розділовими знаками. Розробити програму, яка вилучає із другого тексту всі входження слів першого тексту.

9. Задано два тексти, слова в яких розділені пробілами і розділовими знаками. Розробити програму, яка створює третій текст із слів першого тексту, які не входять у другий текст, розділяючи їх пробілами.

10. Задано два тексти, слова в яких розділені пробілами і розділовими знаками, та окреме слово. Розробити програму, яка після кожного входження заданого слова в перший текст вставляє в нього другий текст.

11. Задано текст, слова в якому розділені пробілами і розділовими знаками, та окремий символ. Розробити програму, яка знаходить і друкує всі слова, в які входить заданий символ найбільшу кількість разів.

12. Задано текст, слова в якому розділені пробілами і розділовими знаками. Розробити програму, яка знаходить і друкує найдовший ланцюжок із слів однакової довжини.

13. Задано текст, слова в якому розділені пробілами і розділовими знаками. Розробити проект програми, яка вилучає з цього тексту всі слова з подвоєнням літер.