

Міністерство освіти і науки України  
Державний університет “Житомирська політехніка”

Кафедра інженерії програмного забезпечення  
Група: ВТ-21-1[1]

Програмування мовою Python  
Лабораторна робота № 4  
«Розгалуження та цикли»

Виконав: Бабушко А. С.  
Прийняв: Морозов Д. С.

					«Житомирська політехніка».21.121.01.000–Лр4						
Змн.	Арк.	№ докум.	Підпис	Дата							
Розроб.		Бабушко А.С.			Звіт з лабораторної роботи			Літ.	Арк.	Аркуші	
Перевір.		Морозов Д.С.							1	00	
Керівник								ФІКТ Гр. ВТ-21-1[1]			
Н. контр.											
Затверд.											

**Мета роботи:** ознайомитися зі рядками в мові Python, діями над ними.

**Хід роботи:**

### **Завдання на лабораторну роботу:**

Завдання 1. Дано список, що складається з N цілочисельних елементів. Список вводиться з клавіатури. Знайти максимальний елемент. Вивести список на екран у зворотному порядку.

### **Лістинг програми:**

```
# task 1
def task_1_output_reverse_list(someList):
    someList.reverse()
    return someList

def enter_some_list():
    print('Enter 0 to end input numbers.')
    newList = []
    while True:
        number = float(input('Enter some number: '))
        if number == 0:
            return newList
        else:
            newList.append(number)

print('\nTASK 1!!!')

task_1_list = enter_some_list()
print(f'Entered list: {task_1_list}')
print(f'Reversed list: {task_1_output_reverse_list(task_1_list)}')
```

### **Результат програми:**

```
TASK 1!!!
Enter 0 to end input numbers.
Enter some number: -2
Enter some number: -1
Enter some number: 1
Enter some number: 2
Enter some number: 3
Enter some number: 4
Enter some number: 5
Enter some number: 6
Enter some number: 0
Entered list: [-2.0, -1.0, 1.0, 2.0, 3.0, 4.0, 5.0, 6.0]
Reversed list: [6.0, 5.0, 4.0, 3.0, 2.0, 1.0, -1.0, -2.0]
```

		Бабушко А.С.			«Житомирська політехніка».21.121.01.000 – Лр4	Арк.
		Морозов Д.С.				2
Змн.	Арк.	№ докум.	Підпис	Дата		

Завдання 2. Дано список, що складається з N цілочисельних елементів. Список вводиться з клавіатури. Переписати всі додатні елементи в другій масив, а решту - в третій.

### Лістинг програми:

```
# task 2
def task_2_get_positive_negative_lists(someList):
    positive_list = []
    negative_list = []
    for number in someList:
        if number >= 0:
            positive_list.append(number)
        else:
            negative_list.append(number)
    return [positive_list, negative_list]

print('\nTASK 2!!!')

task_2_list = enter_some_list()
print(f'Entered list: {task_2_list}')
print(f'Positive list: {task_2_get_positive_negative_lists(task_2_list)[0]}')
print(f'Negative list: {task_2_get_positive_negative_lists(task_2_list)[1]}')
```

### Результат програми:

```
TASK 2!!!

Enter 0 to end input numbers.

Enter some number: -2
Enter some number: -1
Enter some number: 1
Enter some number: 2
Enter some number: 3
Enter some number: 4
Enter some number: 5
Enter some number: 6
Enter some number: 0

Entered list: [-2.0, -1.0, 1.0, 2.0, 3.0, 4.0, 5.0, 6.0]
Positive list: [1.0, 2.0, 3.0, 4.0, 5.0, 6.0]
Negative list: [-2.0, -1.0]
```

Завдання 3. В списку довжиною 20 обчислити суму елементів з непарними індексами. Вивести на екран список, отриману суму.

### Лістинг програми:

```
# task 3
def task_3_get_index_odd_sum_list(someList):
    odd_sum = 0

    for i in range(0, len(someList)):
        if i % 2 != 0:
            odd_sum += someList[i]

    return odd_sum
```

		Бабушко А.С.			«Житомирська політехніка».21.121.01.000 – Лр4	Арк.
		Морозов Д.С.				3
Змн.	Арк.	№ докум.	Підпис	Дата		

```
def get_random_float_number(minRandom, maxRandom):
    import random as random
    return round((random.random() * (maxRandom - minRandom) + minRandom), 2)

def create_random_float_list(n, minRandom, maxRandom):
    newList = []

    for i in range(0, n):
        newList.append(get_random_float_number(minRandom, maxRandom))

    return newList

print('\nTASK 3!!!')
task_3_list = create_random_float_list(20, -5, 5)
print(f'Generated list: {task_3_list}')
print(f'Sum of odd numbers: {task_3_get_index_odd_sum_list(task_3_list)}')
```

### Результат програми:

```
TASK 3!!!
Generated list: [3.85, -1.03, -4.7, 3.89, 1.28, 4.17, 4.68, -2.06, -2.57, -4.76, -2.42, 1.01, -2.82, -3.32, -4.23, -2.37, -3.45, 1.87, -2.72, -3.96]
Sum of odd numbers: -6.559999999999999
```

Завдання 4. Сформувати список з 30 випадкових цілих чисел від -100 до +100. Знайти максимальний елемент списку і його порядковий номер. Отримати інший список, що складається тільки з непарних чисел вихідного списку або повідомити, що таких чисел немає. Отриманий список вивести в порядку зменшення елементів.

### Лістинг програми:

```
# task 4
def get_max_elem_list(some_list):
    return max(some_list)

def get_max_index_list(some_list):
    return some_list.index(max(some_list))

def get_random_int_number(min_random, max_random):
    import random as random
    return round((random.random() * (max_random - min_random) + min_random))

def create_random_int_list(n, min_random, max_random):
    new_list = []

    for i in range(0, n):
        new_list.append(get_random_int_number(min_random, max_random))

    return new_list

def task_5_get_sorted_odd_list(some_list):
    odd_number_list = []

    for number in some_list:
        if number % 2 != 0:
            odd_number_list.append(number)

    odd_number_list.sort(reverse=True)
    return odd_number_list
```

		Бабушко А.С.			«Житомирська політехніка».21.121.01.000 – Лр4	Арк.
		Морозов Д.С.				4
Змн.	Арк.	№ докум.	Підпис	Дата		

```
print('\nTASK 4!!!')
task_4_list = create_random_int_list(30, -100, 100)
print(f'Generated list: {task_4_list}')
print(f'Max value {get_max_elem_list(task_4_list)} found in index {get_max_index_list(task_4_list)}')
print(f'Odd reverse sorted list: {task_5_get_sorted_odd_list(task_4_list)}')
```

### Результат програми:

```
TASK 4!!!
Generated list: [31, -2, -5, -42, -91, 2, -57, -66, 22, -96, 50, 43, 46, -27, -70, 40, 28, -46, 38, 57, 97, 8, -85, -51, -53, 33, -49, -2, 6, 23]
Max value 97 found in index 20
Odd reverse sorted list: [97, 57, 43, 33, 31, 23, -5, -27, -49, -51, -53, -57, -85, -91]
```

Завдання 5. Сформувати список з 30 випадкових цілих чисел від -100 до + 100. Вивести пари від'ємних чисел, що стоять поруч.

### Лістинг програми:

```
# task 5
def task_5_get_negative_pairs(some_list):
    negative_pairs_counter = 0
    for i in range(0, len(some_list) - 1):
        if some_list[i] < 0 and some_list[i + 1] < 0:
            negative_pairs_counter += 1
    print(f'{negative_pairs_counter} {some_list[i]} in {i} and {some_list[i + 1]} in {i + 1}')

print('\nTASK 5!!!')
task_5_list = create_random_int_list(30, -100, 100)
print(f'Generated list: {task_5_list}')
task_5_get_negative_pairs(task_5_list)
```

### Результат програми:

```
TASK 5!!!
Generated list: [-24, -65, 42, -19, -58, 40, -16, -62, -2, -55, -82, -28, 61, -70, -67, -31, -86, 77, -42, 68, -72, 72, -98, 9, 71, -74, -81, -6, -46, 75]
1) -24 in 0 and -65 in 1
2) -19 in 3 and -58 in 4
3) -16 in 6 and -62 in 7
4) -62 in 7 and -2 in 8
5) -2 in 8 and -55 in 9
6) -55 in 9 and -82 in 10
7) -82 in 10 and -28 in 11
8) -70 in 13 and -67 in 14
9) -67 in 14 and -31 in 15
10) -31 in 15 and -86 in 16
11) -74 in 25 and -81 in 26
12) -81 in 26 and -6 in 27
13) -6 in 27 and -46 in 28
```

Завдання 6. Дано список з 10 цілих чисел. Знайти максимальний елемент і порівняти з ним інші елементи. Квадрати менших чисел записати в другий список в порядку зменшення.

### Лістинг програми:

```
# task 6
def task_6_get_squares_list_lower_than_max(some_list):
    squares_list = []
    max_number = max(some_list)

    for number in some_list:
        if number < max_number:
```

		Бабушко А.С.			«Житомирська політехніка».21.121.01.000 – Лр4	Арк.
		Морозов Д.С.				5
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        squares_list.append(number ** 2)

    squares_list.sort(reverse=True)
    return squares_list

print('\nTASK 6!!!')
task_6_list = create_random_int_list(5, -100, 100)
print(f'Generated list: {task_6_list}')
print(f'New list with squared numbers lower than max: {task_6_get_squares_list_lower_than_max(task_6_list)}')

```

### Результат програми:

```

TASK 6!!!
Generated list: [-11, 22, -9, -68, 1]
New list with squared numbers lower than max:
[4624, 121, 81, 1]

```

Завдання 7. Дано список з 30 випадкових цілих і дробових чисел від -100 до + 100. Знайти та вивести мінімальний по модулю елемент. Вивести список на екран в порядку збільшення значення.

### Лістинг програми:

```

# task 7
def task_7_get_min_abs_elem(some_list):
    return abs(min(some_list))

def create_random_int_float_list(n, min_random, max_random):
    import random as random

    new_list = []
    bools = [True, False]

    for i in range(0, n):
        bool_type = bools[random.randint(0, 1)]
        if bool_type:
            new_list.append(get_random_int_number(min_random, max_random))
        else:
            new_list.append(get_random_float_number(min_random, max_random))

    new_list.sort()
    return new_list

print('\nTASK 7!!!')
task_7_list = create_random_int_float_list(30, -100, 100)
print(f'Generated list: {task_7_list}')
print(f'Min abs element in list: {task_7_get_min_abs_elem(task_7_list)}')

```

### Результат програми:

```

TASK 7!!!
Generated list: [-99.56, -89.05, -86, -75.08, -66.74, -54.23, -54, -50.71, -49.1, -49.03, -47, -42, -31, -23.53, -15, -1, 2.03, 11.67, 14, 18, 22.93, 25, 31.44, 36, 41.12, 46, 58, 85.17, 89.52, 89.82]
Min abs element in list: 99.56

```

Завдання 8. Дано список з 30 випадкових цілих і дробових чисел від -100 до + 100. Сформувати зі списку 10 списків по 3 елементи. Вивести отримані списки в порядку зростання за сумою абсолютних значень окремих елементів.

### Лістинг програми:

		Бабушко А.С.			«Житомирська політехніка».21.121.01.000 – Лр4	Арк.
		Морозов Д.С.				
Змн.	Арк.	№ докум.	Підпис	Дата		6

```
# task 8
def task_8_create_3_lists_of_10_elems(some_list):
    for i in range(0, len(some_list)):
        some_list[i] = abs(some_list[i])

    print(some_list)
    first_10_elems = some_list[:10]
    first_10_elems.sort()
    second_10_elems = some_list[10:20]
    second_10_elems.sort()
    third_10_elems = some_list[20:30]
    third_10_elems.sort()

    return [first_10_elems, second_10_elems, third_10_elems]

print('\nTASK 8!!!')
task_8_list = create_random_int_float_list(30, -100, 100)
print(f'Generated list: {task_8_list}')
three_lists_of_10_elems = task_8_create_3_lists_of_10_elems(task_8_list)
print(f'First 10 elems: {three_lists_of_10_elems[0]}')
print(f'Second 10 elems: {three_lists_of_10_elems[1]}')
print(f'Third 10 elems: {three_lists_of_10_elems[2]}')
```

### Результат програми:

```
TASK 8!!!
Generated list: [-92.5, -89.1, -86.14, -83.33, -80, -67.89, -63.62, -51, -46.62, -28, -27.3, -26, -21.63, -10.9, -8.08, -7.09, -7, 3.66, 14, 31.41, 43, 53, 57, 63.67, 70.68, 75, 90.51, 91, 91.18, 94.1]
[92.5, 89.1, 86.14, 83.33, 80, 67.89, 63.62, 51, 46.62, 28, 27.3, 26, 21.63, 10.9, 8.08, 7.09, 7, 3.66, 14, 31.41, 43, 53, 57, 63.67, 70.68, 75, 90.51, 91, 91.18, 94.1]
First 10 elems: [28, 46.62, 51, 63.62, 67.89, 80, 83.33, 86.14, 89.1, 92.5]
Second 10 elems: [3.66, 7, 7.09, 8.08, 10.9, 14, 21.63, 26, 27.3, 31.41]
Third 10 elems: [43, 53, 57, 63.67, 70.68, 75, 90.51, 91, 91.18, 94.1]
```

### Увесь лістинг програми:

```
""" Lab 4. Python. Andrii Babushko. Repository: https://github.com/AndriiBabushko/Python """

# task 1
def task_1_output_reverse_list(someList):
    someList.reverse()
    return someList

def enter_some_list():
    print('Enter 0 to end input numbers.')
    newList = []
    while True:
        number = float(input('Enter some number: '))
        if number == 0:
            return newList
        else:
            newList.append(number)

print('\nTASK 1!!!')

task_1_list = enter_some_list()
print(f'Entered list: {task_1_list}')
print(f'Reversed list: {task_1_output_reverse_list(task_1_list)}')

# task 2
def task_2_get_positive_negative_lists(someList):
    positive_list = []
    negative_list = []
    for number in someList:
        if number >= 0:
```

		Бабушко А.С.			«Житомирська політехніка».21.121.01.000 – Лр4	Арк.
		Морозов Д.С.				7
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        positive_list.append(number)
    else:
        negative_list.append(number)
    return [positive_list, negative_list]

print('\nTASK 2!!!')

task_2_list = enter_some_list()
print(f'Entered list: {task_2_list}')
print(f'Positive list: {task_2_get_positive_negative_lists(task_2_list)[0]}')
print(f'Negative list: {task_2_get_positive_negative_lists(task_2_list)[1]}')

# task 3
def task_3_get_index_odd_sum_list(someList):
    odd_sum = 0

    for i in range(0, len(someList)):
        if i % 2 != 0:
            odd_sum += someList[i]

    return odd_sum

def get_random_float_number(minRandom, maxRandom):
    import random as random
    return round((random.random() * (maxRandom - minRandom) + minRandom), 2)

def create_random_float_list(n, minRandom, maxRandom):
    newList = []

    for i in range(0, n):
        newList.append(get_random_float_number(minRandom, maxRandom))

    return newList

print('\nTASK 3!!!')
task_3_list = create_random_float_list(20, -5, 5)
print(f'Generated list: {task_3_list}')
print(f'Sum of odd numbers: {task_3_get_index_odd_sum_list(task_3_list)}')

# task 4
def get_max_elem_list(some_list):
    return max(some_list)

def get_max_index_list(some_list):
    return some_list.index(max(some_list))

def get_random_int_number(min_random, max_random):
    import random as random
    return round((random.random() * (max_random - min_random) + min_random))

def create_random_int_list(n, min_random, max_random):
    new_list = []

    for i in range(0, n):
        new_list.append(get_random_int_number(min_random, max_random))

    return new_list

```

		Бабушко А.С.			«Житомирська політехніка».21.121.01.000 – Лр4	Арк.
		Морозов Д.С.				8
Змн.	Арк.	№ докум.	Підпис	Дата		



```

def task_5_get_sorted_odd_list(some_list):
    odd_number_list = []

    for number in some_list:
        if number % 2 != 0:
            odd_number_list.append(number)

    odd_number_list.sort(reverse=True)
    return odd_number_list

print('\nTASK 4!!!')
task_4_list = create_random_int_list(30, -100, 100)
print(f'Generated list: {task_4_list}')
print(f'Max value {get_max_elem_list(task_4_list)} found in index {get_max_index_list(task_4_list)}')
print(f'Odd reverse sorted list: {task_5_get_sorted_odd_list(task_4_list)}')

# task 5
def task_5_get_negative_pairs(some_list):
    negative_pairs_counter = 0
    for i in range(0, len(some_list) - 1):
        if some_list[i] < 0 and some_list[i + 1] < 0:
            negative_pairs_counter += 1
            print(f'{negative_pairs_counter} {some_list[i]} in {i} and {some_list[i + 1]} in {i + 1}')

print('\nTASK 5!!!')
task_5_list = create_random_int_list(30, -100, 100)
print(f'Generated list: {task_5_list}')
task_5_get_negative_pairs(task_5_list)

# task 6
def task_6_get_squares_list_lower_than_max(some_list):
    squares_list = []
    max_number = max(some_list)

    for number in some_list:
        if number < max_number:
            squares_list.append(number ** 2)

    squares_list.sort(reverse=True)
    return squares_list

print('\nTASK 6!!!')
task_6_list = create_random_int_list(5, -100, 100)
print(f'Generated list: {task_6_list}')
print(f'New list with squared numbers lower than max:\n{task_6_get_squares_list_lower_than_max(task_6_list)}')

# task 7
def task_7_get_min_abs_elem(some_list):
    return abs(min(some_list))

def create_random_int_float_list(n, min_random, max_random):
    import random as random

    new_list = []
    bools = [True, False]

    for i in range(0, n):
        bool_type = bools[random.randint(0, 1)]
        if bool_type:

```

		Бабушко А.С.			«Житомирська політехніка».21.121.01.000 – Лр4	Арк.
		Морозов Д.С.				9
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        new_list.append(get_random_int_number(min_random, max_random))
    else:
        new_list.append(get_random_float_number(min_random, max_random))

    new_list.sort()
    return new_list

print('\nTASK 7!!!')
task_7_list = create_random_int_float_list(30, -100, 100)
print(f'Generated list: {task_7_list}')
print(f'Min abs element in list: {task_7_get_min_abs_elem(task_7_list)}')

# task 8
def task_8_create_3_lists_of_10_elems(some_list):
    for i in range(0, len(some_list)):
        some_list[i] = abs(some_list[i])

    print(some_list)
    first_10_elems = some_list[:10]
    first_10_elems.sort()
    second_10_elems = some_list[10:20]
    second_10_elems.sort()
    third_10_elems = some_list[20:30]
    third_10_elems.sort()

    return [first_10_elems, second_10_elems, third_10_elems]

print('\nTASK 8!!!')
task_8_list = create_random_int_float_list(30, -100, 100)
print(f'Generated list: {task_8_list}')
three_lists_of_10_elems = task_8_create_3_lists_of_10_elems(task_8_list)
print(f'First 10 elems: {three_lists_of_10_elems[0]}')
print(f'Second 10 elems: {three_lists_of_10_elems[1]}')
print(f'Third 10 elems: {three_lists_of_10_elems[2]}')

```

**Висновок:** під час виконання лабораторної роботи було отримано навички використання списків як структуру даних та деяких методів для роботи з ними.

		Бабушко А.С.			«Житомирська політехніка».21.121.01.000 – Лр4	Арк.
		Морозов Д.С.				10
Змн.	Арк.	№ докум.	Підпис	Дата		