

Лабораторная работа №7

По дисциплине: Основы программной инженерии

Духно Михаил

ПИЖ-б-о-20-1

Задача 1 В заданном список подсчитать число нулевых элементов и вывести на экран их индексы.

Программа task_1.py, код и результат работы

```
lst = input('Write list: ')
lst = lst.split()

num = 0

for ind, elem in enumerate(lst):
    if elem == '0':
        print(f"Zero's index: {ind}")
        num += 1

print(f'Number of elements: {num}')
```

Рисунок 6.1 – Код программы task_1.py

```
C:\Users\я\AppData\Local\Programs\Py
Write list: 0 3 0 -2 4 5 1 0 7 3
Zero's index: 2
Zero's index: 7
Number of elements: 2

Process finished with exit code 0
```

Рисунок 6.2 – Результат работы программы task_1.py

Задача 2: В списке, состоящем из вещественных элементов, вычислить:

- 1. максимальный по модулю элемент списка;*
- 2. сумму элементов списка, расположенных между первым и вторым положительными элементами.*

Программа task_2.py, код и результат работы

```
def find_sum(arr):
    lst_of_elm = []

    for ind, elem in enumerate(arr):
        if float(elem) < 0:
            lst_of_elm.append(ind)
        if len(lst_of_elm) >= 2:
```

```

        break

    sum = 0

    i = lst_of_elm[0] + 1
    while i < lst_of_elm[1]:
        sum = sum + float(arr[i])
        i += 1

    return sum

def find_maximal(arr):
    max = float(arr[0])

    for el in arr:
        elem = float(el)
        if elem < 0:
            elem = elem * (-1)
        if elem > max:
            max = elem
    return max

lst = input('Write list: ')
lst = lst.split()

sum_of_el = find_sum(lst)
max_el = find_maximal(lst)

print(f'Maximal element: {max_el}')
print(f'Sum of elements: {sum_of_el}')

```

Рисунок 6.3 – Код программы task_1.py

```

C:\Users\я\AppData\Local\Programs\Python\Pytho
Write list: 8 3 7 0 18 -3 3 5 1 -9 0 76 11
Maximal element: 76.0
Sum of elements: 9.0

Process finished with exit code 0
|

```

Рисунок 6.4 – Результат работы программы task_1.py

Контрольные вопросы:

1. Что такое списки в языке Python?

Списки в Python - упорядоченные изменяемые коллекции объектов произвольных типов

2. Как осуществляется создание списка в Python?

Присвоить переменной значение [], внутри можно указать список элементов

3. Как организовано хранение списков в оперативной памяти?

Пустой список занимает 72 байта памяти, далее происходит суммирование элементов находящихся в списке

4. Каким образом можно перебрать все элементы списка?

При помощи циклов фор или вайл

5. Какие существуют арифметические операции со списками?

Сложение и вычитание списков

6. Как проверить есть ли элемент в списке?

`x in A`

7. Как определить число вхождений заданного элемента в списке?

`A.count(x)`

8. Как осуществляется добавление (вставка) элемента в список?

`A.append(x)`

9. Как выполнить сортировку списка?

`A.sort()`

При помощи алгоритмов сортировки, например алгоритмом пузырьковой сортировки или сортировки расчёской

10. Как удалить один или несколько элементов из списка?

Метод `remove()` — это встроенный метод, который удаляет первый совпадающий элемент из списка.

Метод `pop()` — удаляет последний элемент списка

Метод `clear()` удаляет все элементы из списка.

11. Что такое списковое включение и как с его помощью осуществлять обработку списков?

Списковое включение — это способ компактного описания операций обработки списков

12. Как осуществляется доступ к элементам списков с помощью срезов?

`Srez = List[3:6]`

13. Какие существуют функции агрегации для работы со списками?

`Len()`

`Max()`

`Min()`

`Enumerate()`

14. Как создать копию списка?

```
newlist = list.copy()
```

15. Самостоятельно изучите функцию `sorted` языка Python. В чем ее отличие от метода `sort` списков?

Функция `sorted` возвращает новый отсортированный список, который получен из итерируемого объекта, который был передан как аргумент. Функция также поддерживает дополнительные параметры, которые позволяют управлять сортировкой.

`sorted()` возвращает **новый** отсортированный список, оставляя исходный список незатронутым. `list.sort()` сортирует список **на месте**