

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
ИНСТИТУТ ЦИФРОВОГО РАЗВИТИЯ

Отчет о лабораторной работе №7 по дисциплине «Основы
программной инженерии»

Выполнил:
Прокопов Дмитрий
Владиславович,
2 курс, группа ПИЖ-б-о-21-1,
Проверил:
Доцент кафедры инфокоммуникаций,
Воронкин Р.А.

Ставрополь, 2022 г

Ход работы

```
#!/usr/bin/env python3
#-*- coding: utf-8 -*-
import sys
if __name__=='__main__':
    A=list(map(int, input().split()))
    if len(A)!=10:
        print("Неверный размер списка", file=sys.stderr)
        exit(1)
    s=sum([a for a in A if abs(a)<5])
    print(s)

if __name__=='__main__':
    primer1
```

C:\Users\dmidt\PycharmProjects\7lb\venv\Scripts\python.exe C:/

4 12 53 7 33 1 98 6 31 6

5

Рис. 7.1 – выполнение 1-го примера

```
#!/usr/bin/env python3
#-*- coding: utf-8 -*-
import sys
if __name__=='__main__':
    a=list(map(int, input().split()))
    if not a:
        print("Заданный список пуст", file=sys.stderr)
        exit(1)
    a_min=a_max=a[0]
    i_min=i_max=0
    for i, item in enumerate(a):
        if item<a_min:
            i_min, a_min=i, item
        if item>=a_max:
            i_max, a_max= i, item
    if i_min > i_max:
        i_min, i_max = i_max, i_min
    count=0
    for item in a[i_min+1:i_max]:
        if item>0:
            count+=1
    print(count)
```

Рис. 7.2 – 2-й пример

```
C:\Users\dmidt\PycharmProjects\7lb\venv\
Заданный список пуст
Process finished with exit code 1
```

Рис. 7.3 – вывод если список пуст

```
C:\Users\dmidt\PycharmProjects\7lb\venv\Scripts\python
42 62 7 74 21 44 76 88 21 9
4
Process finished with exit code 0
```

Рис. 7.4 – выполнение 2-го примера

```
#!/usr/bin/env python3
#-*- coding: utf-8 -*-
if __name__ == '__main__':
    A=list(map(int, input("Введите оценки по алгебре\n").split()))
    G=list(map(int, input("Введите оценки по геометрии\n").split()))
    F=list(map(int, input("Введите оценки по физике\n").split()))
    a=g=f=0
    for i in range(len(A)):
        if A[i]>G[i] and A[i]>F[i]:
            a+=1
        if G[i]>A[i] and G[i]>F[i]:
            g+=1
        if F[i]>A[i] and F[i]>G[i]:
            f+=1
    print("Кол-во оценок по алгебре лучше:", a)
    print("Кол-во оценок по геометрии лучше:", g)
    print("Кол-во оценок по физике лучше:", f)
    if a>g and a>f:
        print("Лучшие оценки по алгебре")
    if g>a and g>f:
        print("Лучшие оценки по геометрии")
    if f>a and f>g:
        print("Лучшие оценки по физике")
```

Рис. 7.5 – 1-е индивидуальное задание

```

Введите оценки по алгебре
4 5 3 5 4 3 5
Введите оценки по геометрии
4 3 3 4 5 3 4
Введите оценки по физике
5 5 3 4 5 4 5
Кол-во оценок по алгебре лучше: 1
Кол-во оценок по геометрии лучше: 0
Кол-во оценок по физике лучше: 2
Лучшие оценки по физике

```

Рис. 7.6 – выполнение программы

```

#!/usr/bin/env python3
# -*- coding: utf-8 -*-
if __name__ == '__main__':
    A=list(map(int, input("Введите целые элементы\n").split()))
    p, s, k1, k2=1, 0, 0, 0
    for i in range(len(A)):
        if i%2==0:
            p*=A[i]
        if A[i]==0:
            if k1==0:
                k1=i
            k2=i
    print("Произведение элементов списка с четными номерами:", p)
    for i in range(len(A)):
        if A[i]>k1 and A[i]<k2:
            s+=A[i]
    print("Сумма элементов между первым и последним 0 элементом:", s)
    so=sorted(list(filter(lambda x: x>=0, A)))+sorted(list(filter(lambda x: x<0, A)))
    print(so)

```

Рис.7.7 – 2-я индивидуальная работа

```

Введите целые элементы
64 28 -4 0 34 8 -2 5 7 -53 165 0 83 64 -4
Произведение элементов списка с четными номерами: -6675271680
Сумма элементов между первым и последним 0 элементом: 20
[0, 0, 5, 7, 8, 28, 34, 64, 64, 83, 165, -53, -4, -4, -2]

```

Рис. 7.8 – выполнение 2-ой инд. работы
 Ответы на вопросы:

1. Список (list) – это Для создания списка нужно заключить элементы в квадратные скобки.
2. В оперативной памяти списки хранятся в виде ссылок, в которых хранятся ссылики на другие элементы.
3. Перебрать элементы списка можно с помощью цикла.
4. Сложение и умножение.
5. Для проверки принадлежности элемента списку существует оператор in

6. `s.count('')`
7. `s.insert(1,'')` – Вставить после заданного индекса, `s.append('')` – Вставить в конец списка.
8. `s.sort()`
9. Для удаления можно использовать:
 - a. `s.pop(index)`
 - b. `s.remove('element')`
 - c. `del s[0]`, `del s[1:3]`
10. Абстракция списков или списковое включение, является частью синтаксиса языка, которая предоставляет простой способ построения списков.
11. Срезы списков работают одинаково со срезами строк: `s[start:stop:step]`
12. Функции агрегации:
 - a. `len(s)` – число элементов в списке
 - b. `min(s)` – минимальный элемент списка
 - c. `max(s)` – максимальный элемент списка
 - d. `sum(s)` – сумма элементов списка
13. `es = s.copy()`
14. `s.sort()` сортирует исходный список, `sorted` возвращает отсортированный список.
15. структура данных для хранения объектов различных типов