МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное образовательное учреждение высшего образования «Санкт-петербургский государственный морской технический университет»

ФАКУЛЬТЕТ ЦИФРОВЫХ ПРОМЫШЛЕННЫХ ТЕХНОЛОГИЙ

Кафедра Киберфизических систем

ЛАБОРАТОРНАЯ РАБОТА

По дисциплине «Программирование»

Выполнил: Коткин Вадим Иванович

Проверил: Поделенюк Павел Петрович

Санкт-Петербург

2024

Оглавление

[1. Цели и формулировка задачи 3](#_Toc187676633)

[2. Результаты работы 4](#_Toc187676634)

[2.1. Реализация программы с использованием функционального программирования языка Python 4](#_Toc187676635)

[2.1.1. Ход работы 4](#_Toc187676636)

[2.1.2. Демонстрация работы программы 4](#_Toc187676637)

[2.1.3. Листинг кода 4](#_Toc187676638)

[ЗАКЛЮЧЕНИЕ 5](#_Toc187676639)

[Список использованных источников 6](#_Toc187676640)

# Цели и формулировка задачи

ЦЕЛИ РАБОТЫ  
При работе над данной лабораторной работой будут затронуты следующие вопросы:  
1. Работа с файлами,  
2. Работа с несколькими функциями, а не только с main,   
3. Реализации рекурсивной функции.  
  
Формулировка задания  
Даны N целых чисел X1, X2, …, XN. Требуется расставить между ними знаки + и - так, чтобы значение получившегося выражения было равно заданному целому S.  
Входные данные: считываются с файла через пробел, в следующем порядке:   
N X1 X2 ... XN S  
2 <= N <= 30, 0 <= Xi <= 5\*107, -109 <= S <= 109  
Необходимо вывести в файл полученное равенство с расставленными знаками, либо no solution, если не найдено решения. Если найдено несколько решений, то необходимо вывести любое. Решение, только через рекурсию!  
Пример:  
Input:  
 3 2 3 4 3  
Output:  
2-3+4=3

# Результаты работы

## **Реализация программы с использованием функционального программирования языка Python**

## Ход работы

Этот код предназначен для поиска арифметического выражения, состоящего из элементов, указанных в файле, которые при помощи операций сложения и вычитания в сумме дают определенное число. Вот ход работы кода:  
  
1. Функция read\_file: Открывает файл, считывает первую строку, разбивает ее на составляющие и возвращает список значений.   
  
2. Функция f: Рекурсивно генерирует все возможные арифметические выражения, используя элементы списка:  
   - Проверяет, достигнут ли конец списка.  
   - Если сумма текущих элементов (csum) равна заданной сумме (tsum), возвращает выражение.  
   - В противном случае вызывает себя дважды: один раз, добавляя следующий элемент с плюсом, и второй раз — с минусом.  
  
3. Функция main: Основная логика:  
   - Считывает данные из файла с помощью read\_file.  
   - Разделяет полученные данные для корректной обработки.  
   - Вызывает функцию f, чтобы найти нужное выражение.  
   - Записывает результат в файл "output.txt". Если решение найдено, записывает выражение и ожидаемую сумму. Если нет — сообщение "No solution".

4. Запуск программы:

- выполняет main

## Демонстрация работы программы

## 

## Листинг кода

1. def read\_file(file\_path):  
    with open(file\_path, 'r') as h:  
    ent = h.readline().strip().split()  
    return ent  
     
   def f(a, index, tsum, csum, expr):  
    if index == len(a):  
    if csum == int(tsum):  
    return expr  
    return None  
     
    result = f(a, index + 1, tsum, csum + int(a[index]), expr + '+' + a[index] if index > 0 else a[index])  
     
    if result:  
    return result  
     
    result = f(a, index + 1, tsum, csum - int(a[index]), expr + '-' + a[index])  
    return result  
     
   def main():  
    global f  
    ent = read\_file('input.txt.txt')  
    a = ent[1:len(ent) - 1]  
    tsum = ent[-1]  
    b = f(a, 0, tsum, 0, '')  
    if b:  
    with open('output.txt', 'w') as f:  
    f.write(b + '=' + tsum)  
    else:  
    with open('output.txt', 'w') as f:  
    f.write('No solution')  
     
   if \_\_name\_\_ == "\_\_main\_\_":  
    main()

# ЗАКЛЮЧЕНИЕ

В ходе работы мы научились работать с файлами. Научились работать с несколькими функциями, а не только с main. И научились реализовывать рекурсивную функцию.

# 

# Список использованных источников