Министерство науки и высшего образования Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ МОРСКОЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

(СПБГМТУ)

|  |
| --- |
|  |

Отчёт по Лабораторной работе №1

|  |
| --- |
|  |

|  |
| --- |
| Выполнил студент группы 20121: Никитин Д В. |
| Проверил: Поделенюк П.П. |

|  |
| --- |
|  |

Санкт-Петербург  
2023год

# 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.

# ХОД РАБОТЫ

Программа считывает данные из файла и сохраняет их в виде списка в переменной input\_data. Далее переменной N передаётся первое значение из input\_data – количество целых чисел в выражении. С помощью метода pop() это значение удаляется из input\_data. Переменной S передаётся последнее значение из input\_data – значение, которому должно быть равно выражение. Это значение тем же методом удаляется из input\_data. Вводим переменную signs, которая равна пустому списку и в которой мы далее будем хранить знаки между числами в выражении. С помощью спискового выражения мы передаём переменной sequence числовые значения нашего выражения. Это было необходимо так как, в input\_data значения были не int, а str.

С помощью метода write() мы записываем в файл результат выполнения функции output(), но чтобы понять, что делает эта функция, необходимо сперва разобраться с функцией sign\_tree(). Эта функция принимает последовательность (seq: list) и значение (goal: int), которое должно принимать выражение, а возвращает порядок знаков или пустой список, если решения нет. В начале мы делаем переменную signs глобальной, чтобы была возможность её изменять. Далее идет проверка на количество элементов в списке. Если элемент всего один, то мы проверяем, равняется ли он goal, если да, то возвращаем True, иначе False. Если элементов больше чем 1, то мы, с помощью метода copy(), передаем в переменные for\_plus и for\_minus копии последовательности seq. Следующим шагом мы изменяем значения 1-ого элемента for\_plus путём прибавления к нему 0-ого элемента. Из значения 1-ого элемента for\_minus мы вычитаем 0-ой элемент. Далее мы проверяем, является ли sign\_tree(), с переданным в качестве последовательности for\_plus с первого до последнего элемента, True. Если да, то мы добавляем в signs ‘+’ и возвращаем signs, тем самым останавливая выполнение функции. Благодаря этому создаётся рекурсия. Следующей проверкой мы делаем то же самое, но передаём уже for\_minus[1:] и если условие выполняется, то добавляем ‘-’, а не ‘+’.

Теперь можно обсудить функцию output(). Своими аргументами они принимает последовательность (seq: list), список знаков (signs: list) и значение (goal: int), которое должно принимать выражение, а возвращает уравнение или ‘no solution’. Сперва мы проверяем, не пуста ли signs. Если список пустой, то возвращает ‘no solution’. Если нет, то задаём с помощью f-строк значение переменной equation в виде первого элемента seq, далее необходимо перевернуть signs, так как знаки каждый раз добавлялись в конец списка и находятся не в том порядке. С помощью цикла и f-строк добавляем в переменную знак и следующий за ним элемент выражения. В конце добавляется знак ‘=’ и goal. Возвращаем equation.

После того, как мы всё записали в файл, его необходимо закрыть с помощью метода close().

# РЕЗУЛЬТАТ РАБОТЫ

Для заданного в файле ‘file.txt’  
  
Будет вывод в файле ‘file.txt’  


# ВЫВОД

В результате работы цели были достигнуты в решении присутствует: работа с файлами, работа с несколькими функциями, а не только с main, реализации рекурсивной функции.

# Листинг кода

