Министерство образования и науки Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования   
Санкт-Петербургский Государственный Морской Технический Университет

Лабораторная работа № 1

По дисциплине: «Программирование»

|  |  |
| --- | --- |
| Выполнил: | Расщупкин Дмитрий Антонович, студент СПБГМТУ, группа 20121 |
| Принял: | Ассистент СпБГМТУ Поделенюк Павел Петрович |

Санкт-Петербург

2023

Оглавление

[Цели работы 3](#_Toc122882636)

[Ход работы 4](#_Toc122882637)

[Результат работы 5](#_Toc122882638)

[Вывод 6](#_Toc122882639)

[Листинг кода 7](#_Toc122882640)

# 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

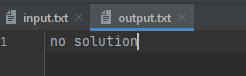
1. Считывается файл input и получаем количество чисел N, итоговую сумму S и список чисел X. Дальше преобразуем в нужный нам вид.
2. Дальше вызываем нашу рекурсивную функцию, в которой и происходит расстановка знаков «+» и «-».
3. Сперва проверяем условие индекса, чтобы не выйти за пределы списка, дальше проверяем временную сумму чисел.
4. Дальше написаны две проверки для расстановки знаков «+», «-», логика одинакова только различается в знаках «+» и «-». В каждой функции написаны условие под соответствующий знак. Записывается знак во временный массив знаков temp, изменяется временная сумма sum, вызывается рекурсивная функция, восстанавливается временная сумма и проверяется условие, которое зависит от 3 пункта (первого условия).
5. Если функция возвращает True, то в файл записываем “no solution”. Если функция возвращает False, то в файл записывается из списка чисел и списка знаков готовый ответ.

# Результат работы

Пример ввода и вывода, при N = 6; x1, …, xN = [1, 2, 3, 4, 5, 6]; S = 25



Пример ввода и вывода, при N = 6; x1, …, xN = [3, 2, 3, 4, 5, 6]; S = 21



# Вывод

По итогу выполнения данной лабораторной работы улучшились и закрепились навыки работы с

1. файлами,
2. несколькими функциями, а не только с «main»,
3. Реализацией рекурсивной функции.

# Листинг кода

**def** **recursion**(index: int, N: int, S: int, sum: int, X: list, res: list, temp: list) -> bool:

**if** index == N - **1**: # условие, которое помогает нам выйти из рекурсии

**if** sum == S: # с помощью отслеживания индекса итерации

**return** False # помогает завершить итерацию рекурсии и закончить работу рекрсивной функции

**else**:

**return** True

**if** sum - X[index + **1**] + res[index + **2**] >= S: # условие, которое помогает нам установить знак "-"

temp[index] = '-' # добавляем в список знаков "-"

sum -= X[index + **1**] # изменение суммы (отнимаем следующее число)

solution = recursion(index + **1**, N, S, sum, X, res, temp) # вход в рекурсию и увеличение индекса на единцу, что помогает не сделать бесконечную рекурсию

sum += X[index + **1**] # добавляем к сумме то, что отнимали

**if** **not** solution:

**return** False # если на какой-либо итерации код достиг False (первое условие сработало), то выход из итерации

**if** sum + X[index + **1**] - res[index + **2**] <= S: # условие, которое помогает нам установить знак "+"

temp[index] = '+' # добавляем в список знаков "+"

sum += X[index + **1**] # изменение суммы (прибавлем следующее число)

solution = recursion(index + **1**, N, S, sum, X, res, temp) # вход в рекурсию и увеличение индекса на единцу, что помогает не сделать бесконечную рекурсию

sum -= X[index + **1**] # отнимаем из суммы то, что прибавляли

**if** **not** solution:

**return** False # если на какой-либо итерации код достиг False (первое условие сработало), то выход из итерации

**return** True # алгоритм (функция не нашла подходящей комбинации знаков для получении нужной суммы

**def** **main**():

f = open("input.txt").readline().split(' ') # открытие файла с инпутом

N, S = int(f[**0**]), int(f[-**1**]) # определение кол-ва чисел и нужной нам суммы

X = [] # пустой массив для хранения нужных нам чисел

**for** i **in** range(**0**, len(f)):

**if** i == **0** **or** i == len(f) - **1**: # цикл для заполнения Х

**continue**

X.append(int(f[i]))

res = list(X)

res.append(**0**) # добавление 0 для последующей правильной работы цикла

**for** i **in** range(N - **1**, -**1**, -**1**):

res[i] += res[i + **1**]

temp = [**0**] \* N # создание переменной длины N для хранения наших знаков

sum = X[**0**]

solution = recursion(**0**, N, S, sum, X, res, temp) # использование рекурсивной функции по параметрам: индекса, N, S, sum, res, temp

**if** solution == **0**: # если рекурсивная функция вернула False (0), то мы записываем ответ в файл

text = str(X[**0**]) # записываем первое число из списка данных

**for** i **in** range(**0**, N - **1**):

text += str(temp[i]) + str(X[i + **1**]) # добавляем знак к готовому ответу и последующую цифру из списка

text += "="+str(S) # добавление знака "="

file = open("output.txt", "w") # запись в файл ответа

file.write(text)

file.close()

**else**: # если True, то записываем no solution

file = open("output.txt", "w")

file.write('no solution')

file.close()

**if** \_\_name\_\_ == '\_\_main\_\_':

main()