МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное

учреждение высшего образования

«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

|  |
| --- |
| Институт информационных технологий |
| Кафедра математического и программного обеспечения |
| Структуры и алгоритмы обработки данных |
|  |

ЛАБОРАТОРНАЯ РАБОТА №3

|  |  |
| --- | --- |
| Исполнитель: | студент  группы 1ПИб-02-3оп-23  Богданов  Ренат Алексеевич |
| Руководитель: | Журавлёва Юлия Михайловна |
| Оценка: |  |
| Подпись: |  |

2024 год

Задача

Сумма чисел

В текстовом файле имеется N (N <=20) натуральных чисел по одному в каждой строке. Расставить между этими числами знаки "+" и "-" так, чтобы в результате вычислений получилось число А, запрашиваемое с клавиатуры. Программа должна находить все варианты расстановки знаков, подсчитывать их количество и выводить результаты в текстовый файл.

Описание работы программы.

Работа программы состоит из следующих этапов:

1. **Чтение данных из файла**: программа считывает числа из текстового файла data.txt и сохраняет их в вектор numbers.

2. **Ввод целевого числа**: пользователь вводит число A, которое необходимо получить в результате вычислений.

3. **Перебор комбинаций**: программа генерирует все возможные комбинации знаков "+" и "-" для n-1 операторов (где n - количество чисел в файле). Это реализовано с помощью битовых операций, где каждый бит в двоичном представлении числа соответствует знаку оператора: 0 - "+", 1 - "-".

4. **Вычисление суммы**: для каждой комбинации знаков программа вычисляет сумму чисел, используя соответствующие знаки.

5. **Проверка на совпадение**: проверяется, равна ли полученная сумма заданному числу A.

6. **Запись результата**: если сумма совпадает с A, комбинация знаков записывается в файл result.txt.

7. **Подсчет вариантов**: программа подсчитывает общее количество найденных вариантов.

8. **Вывод результата**: количество найденных вариантов выводится в файл result.txt.

Код программы

#include <iostream>

#include <fstream>

#include <vector>

int main() {

setlocale(LC\_ALL, "RU");

std::ifstream file("data.txt");

// Число-сумма для нахождения

int A;

// Текущее число, взятое из файла для добавления в массив

int number;

// Массив чисел из файла

std::vector<int> numbers;

// Массив из 0 и 1 для операторов "+" и "-"

std::vector<int> op;

// Чтение файла и запись чисел в массив

while (file >> number) {

numbers.push\_back(number);

}

// Запрос числа A

std::cout << "Введите число A: ";

std::cin >> A;

// Вывод содержимого массива numbers (и, соответственно, содержимого файла)

std::cout << "Числа из файла data.txt:" << std::endl;

for (int i = 0; i < numbers.size(); i++) {

std::cout << numbers[i] << " ";

}

std::cout << std::endl;

// Заполнение массива операторов

for (int i = 0; i < numbers.size() - 1; i++) {

op.push\_back(0);

}

std::cout << std::endl;

// Перебор всех комбинаций операторов

int count = 0;

std::ofstream resultFile("result.txt");

for (int i = 0; i < (1 << op.size()); i++) {

// Запись текущей комбинации операторов в массив op

for (int j = 0; j < op.size(); j++) {

op[j] = (i >> j) & 1;

}

// Вычисление суммы для текущей комбинации

int sum = numbers[0];

for (int j = 0; j < op.size(); j++) {

if (op[j] == 0) {

sum += numbers[j + 1];

}

else {

sum -= numbers[j + 1];

}

}

// Проверка на совпадение суммы с A

if (sum == A) {

count++;

// Вывод результата в файл

resultFile << numbers[0];

for (int j = 0; j < op.size(); j++) {

if (op[j] == 0) {

resultFile << " + " << numbers[j + 1];

}

else {

resultFile << " - " << numbers[j + 1];

}

}

resultFile << " = " << A << std::endl;

}

}

// Вывод количества найденных комбинаций

std::cout << "Найдено " << count << " вариантов." << std::endl;

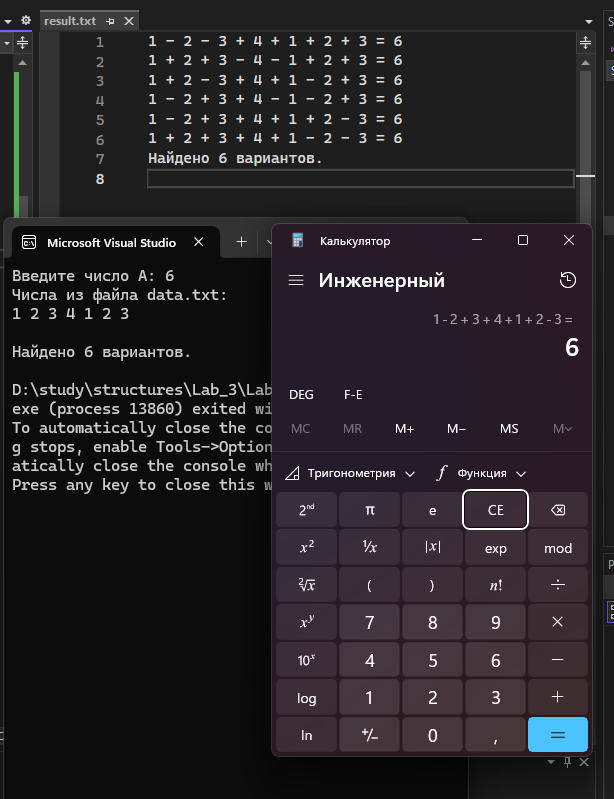
resultFile << "Найдено " << count << " вариантов." << std::endl;

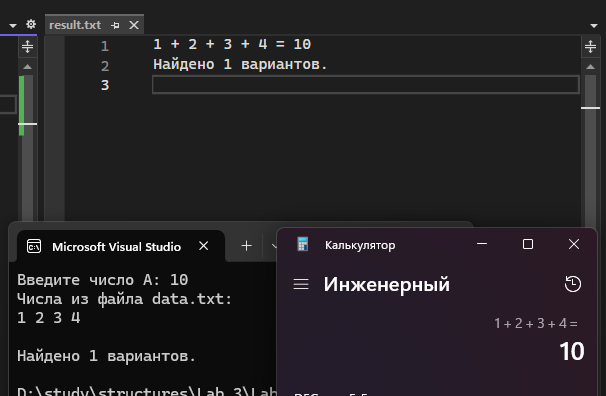
resultFile.close();

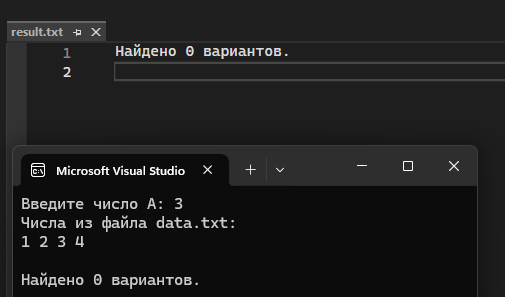
return 0;

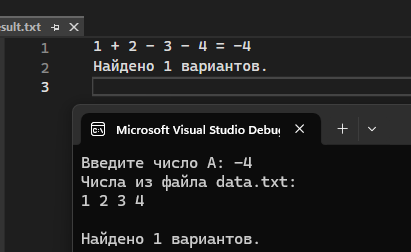
}

Примеры работы программы (с выборочной проверкой









Вывод

В ходе выполнения лабораторной работы были освоены битовые операции, рекурсивный алгоритм, работа с файлами и стандартными контейнерами C++.