ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчёт по лабораторной работе № 4

«Название лабораторной работы)»

Выполнил работу

Дьячкова Елизавета

Академическая группа J3112

Принято

Практикант, Максим Дунаев

Санкт-Петербург

2024

Структура отчёта:

**1. Введение**

Цель: Целью данной работы является проверка возможности разделения массива на две части с равными суммами.

Задачи:

* Реализовать решение с помощью рекурсивной функции, которая помогает разделить массив на две группы таким образом, чтобы суммы элементов в этих группах были равны, перебирая способы распределения элементов массива по двум группам
* Оценить и проанализировать сложность этого алгоритма

**2. Теоретическая подготовка**

**Типы данных:**

Массивы и векторы:

* **vector<int>** - динамический массив
* **a** – массив для хранения исходных элементов
* **tek1** и **tek2** – для текущих групп
* **ans1** и **ans2** – для хранения решений

Переменные:

* **n, sm1** и **sm2, cnt**

**Алгоритм:**

Использован рекурсивный метод решения, который позволяет перебрать все возможные варианты распределения элементов массива по двум группам. Этот подход требует перебора всех подмножеств массива с возможностью разделения.

Рекурсивный алгоритм в данной задаче заключается в том, чтобы для каждого элемента массива на каждом шаге рекурсии попробовать добавить его либо в первую группу, либо во вторую. Когда мы рассмотрим все элементы массива, если суммы обеих групп окажутся равными, то это будет решением задачи.

**3. Реализация**

**1 Этап.** **Подготовка структуры данных и инициализация**

Сначала была подготовлена структура данных для хранения массива и временных групп, в которые элементы массива будут добавляться в процессе работы алгоритма:

**vector<int> a, tek1, tek2, ans1, ans2**

и задание массива, чтобы протестировать программу:

**vector<int> arr = {1, 5, 11, 5};**

**n = arr.size();**

**a = arr;**

Но массив может быть введен пользователем:

**cin >> n; a.resize(n);**

**for (int i = 0; i < n; ++i)**

**cin >> a[i];**

**2 Этап. Реализация Рекурсивной функции**

Следующим шагом была разработка рекурсивной функции **f**, которая является основой алгоритма. Она рекурсивно пытается разделить элементы массива на две группы с равными суммами.

Её параметры:

* **sm1 и sm2** -  суммы элементов в первой и второй группах на текущем шаге рекурсии
* **cnt** - индекс текущего элемента в массивe

**3 этап. Запуск рекурсии и получение результата**

**f(0, 0, 0); -** вызов функции с начальной суммой для обеих групп равной 0 и начальным индексом **сnt** = 0, алгоритм начинает с первого элемента массива

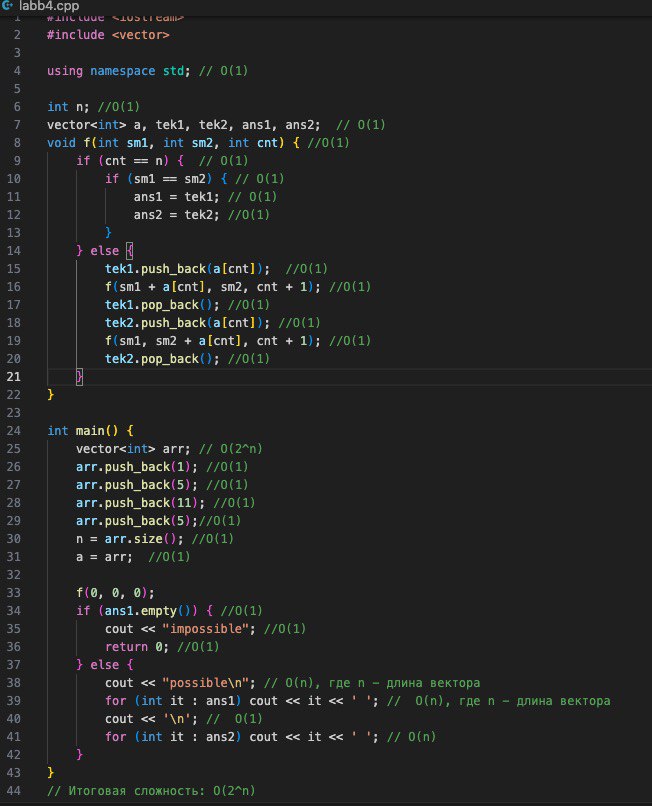
**4 этап. Вывод результата**

Проверка нахождения групп с равными суммами, если вектор **ans1** пустой, то выводится **“impossible”** и решение не найдено, а если **ans1** не пуст, то решение найдено и выводится **“possible”** с элементами групп

Особенности реализации заключаются в использовании рекурсии, проверке на равенство сумм и использовании vector.

**4. Экспериментальная часть**

* подсчет по памяти
* подсчет асимптотики



Таким образом, итоговая сложность равна O(2^n)

**5. Заключение**

В ходе выполнения работы мною был реализован рекурсивный алгоритм, который дал возможность разделить массив на две части с равными суммами. Цель работы была достигнута с помощью рекурсивной функции. В качестве дальнейших исследований можно предложить оптимизацию алгоритма с точки зрения уменьшения затрат использования памяти

**6. Приложения**

