

Отчет

по лабораторной работе «1005. Куча камней»

по дисциплине «Алгоритмы и Структуры данных»

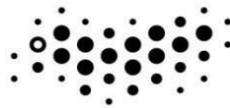
Авторы:

Полит Алексей Денисович

Факультет: СУиР

Группа: R3235

Преподаватель: Тропченко Андрей Александрович



УНИВЕРСИТЕТ ИТМО

1. Задача

У вас есть несколько камней известного веса w_1, \dots, w_n . Напишите программу, которая распределит камни в две кучи так, что разность весов этих двух куч будет минимальной.

2. Исходные данные

Ввод содержит количество камней n ($1 \leq n \leq 20$) и веса камней w_1, \dots, w_n ($1 \leq w_i \leq 100\,000$) — целые, разделённые пробельными символами.

3. Текст программы

```
#include <iostream>
using namespace std;

int search(int* array, int size, int i = 0, int sum1 = 0, int sum2 = 0) {
    static int min_ = 100000;
    if (i == size) min_ = min(min_, abs(sum1 - sum2));
    else {
        search(array, size, i + 1, sum1 + array[i], sum2);
        search(array, size, i + 1, sum1, sum2 + array[i]);
    }
    return min_;
}

int main() {
    int n;
    cin >> n;
    int rocks[n];
    for (int i = 0; i < n; i++) cin >> rocks[i];
    cout << search(rocks, n) << endl;
    return 0;
}
```

ID	Дата	Автор	Задача	Язык	Результат проверки	№ теста	Время работы	Выделено памяти
9259369	11:49:54 9 мар 2021	Aleksey	1005. Куча камней	G++ 9.2 x64	Accepted		0.015	404 КБ

4. Описание программы

Нам нужно минимизировать разницу веса между двумя кучами камней. Первое что приходит - разделить исходную кучу на 2 и посчитать суммы. Очевидно, что метод не даст правильного результата, так как в одной половине могут лежать тяжёлые, а в другой лёгкие камни. Но, подобная куча может нам подойти. Теперь попробуем класть в каждую кучу пары из самого тяжёлого и самого лёгкого камня в исходной куче. Ручная

трассировка алгоритма на тестовом множестве показала, что вариант гиблый.

Суммы между весом n -го камня и $n + 1$ камня в отсортированный по убыванию куче, может оказаться значительно больше, чем у следующей пары. Получается, нам нельзя брать камни из противоположных концов множества, нужно брать их строго по порядку, чтобы в каждой из куч был максимально тяжёлый не взятый камень. А для минимизации разрыва весов с каждым взятым камнем, будем действовать так: возьмём камень $p1$ для левой кучи. Возьмём камень $p2$ для правой кучи. Возьмём камень $p3$ и положим его в меньшую из куч. С каждым последующим камнем мы будем поступать похожим образом.

Будем на каждом шаге рекурсии класть камешек в разные кучи, а на вершине стека сравним модуль полученной разности куч (обязательно модуль! так-как в отличие от цикла мы не задаём явно, в какую кучу класть камень при их равенстве) с заведомо большим значением.