

Отчет

по лабораторной работе «1155. Дуоны»

по дисциплине «Алгоритмы и Структуры данных»

Авторы:

Полит Алексей Денисович

Факультет: СУиР

Группа: R3235

Преподаватель: Тропченко Андрей Александрович



УНИВЕРСИТЕТ ИТМО

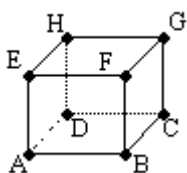
1. Задачи

Архангел по науке докладывает:

— Господи, эти физики там, внизу, — они открыли ещё одну элементарную частицу!

— Хорошо, добавим параметр в Общее Уравнение Вселенной.

С развитием техники физики находят всё новые и новые элементарные частицы, с непонятными и даже загадочными свойствами. Многие слышали про мюоны, глюоны, странные кварки и прочую нечисть. Недавно были обнаружены элементарные частицы дуоны. Эти частицы названы так потому, что учёным удаётся создавать или аннигилировать их только парами. Кстати, от дуонов одни неприятности, поэтому от них стараются избавляться до начала экспериментов. Помогите физикам избавиться от дуонов в их установке.



Экспериментальная установка состоит из восьми камер, которые расположены в вершинах куба. Камеры промаркированы латинскими буквами A, B, C, ..., H. Технически возможно создать, или наоборот, аннигилировать, два дуона, находящихся в смежных камерах. Вам нужно автоматизировать процесс удаления дуонов из установки

2. Исходные данные

В единственной строке даны восемь целых чисел в пределах от 0 до 100, описывающих количество дуонов в камерах установки (сначала в камере A, потом в B, и т.д.).

3. Текст программы

```

#include <iostream>
using namespace std;
struct Point{
    char ch;
    int n;
};
void minus_p(Point* x, Point* y) {
    --(x->n);
    --(y->n);
    cout << y->ch << x->ch << '-' << endl;
};
void plus_p(Point* x, Point* y) {
    ++(x->n);
    ++(y->n);
    cout << y->ch << x->ch << '+' << endl;
};
int main() {
    Point a = {'A'}, b = {'B'}, c = {'C'}, d = {'D'}, e = {'E'}, f = {'F'}, g = {'G'}, h = {'H'};
    cin >> a.n >> b.n >> c.n >> d.n >> e.n >> f.n >> g.n >> h.n;
    if (a.n + c.n + f.n + h.n != b.n + d.n + e.n + g.n) {
        cout << "IMPOSSIBLE" << endl;
        return 0;
    }
    while (a.n + c.n + f.n + h.n > 0) {
        if (a.n > 0) {
            if (b.n > 0) minus_p (&a, &b);
            else if (d.n > 0) minus_p(&a, &d);
            else if (e.n > 0) minus_p(&a, &e);
            else if (g.n > 0) {
                plus_p(&f,&b);
                minus_p (&a, &b);
            }
        }
        else if (h.n > 0) {
            if (g.n > 0) minus_p(&h, &g);
            else if (e.n > 0) minus_p(&h, &e);
            else if (d.n > 0) minus_p(&h, &d);
            else if (b.n > 0) {
                plus_p(&d,&c);
                minus_p(&h, &d);
            }
        }
        else if (f.n > 0) {
            if (b.n > 0) minus_p(&f, &b);
            else if (e.n > 0) minus_p(&f, &e);
            else if (g.n > 0) minus_p(&f, &g);
            else if (d.n > 0) {
                plus_p(&a,&b);
            }
        }
    }
}

```

```

        minus_p(&f, &b);
    }
}
else if (c.n > 0) {
    if (g.n > 0) minus_p(&c, &g);
    else if (d.n > 0) minus_p(&c, &d);
    else if (b.n > 0) minus_p(&b, &c);
    else if (e.n > 0) {
        plus_p(&f, &b);
        minus_p(&b, &c);
    }
}
}
}
return 0;
}

```

9276930

12:55:52
22 мар 2021

[Aleksy](#)

[1155. Дуоны](#)

G++ 9.2 x64

Accepted

0.001

400 КБ

4. Описание программы

Разобьём вершины куба на 2 группы несмежных вершин: A-C-F-H и B-D-E-G. Задача не решится в случае, когда сумма в вершинах не совпадает. Для хранения используем структуру, которая хранит имя и её вес. Используя цикл будем проверять возможности проводить аннгиляцию со смежными вершинами. В случае, если есть вершина из другой группы, которая не смежная и имеющая вес, то добавим вес двум другим вершинам, которые соединяют эту вершину с вершиной из другой группы.