

*Национальный исследовательский университет ИТМО   
(Университет ИТМО)*

*Факультет систем управления и робототехники*

Дисциплина: Алгоритмы и структуры данных

**Отчет по практической работе (2025 задача).**

Студент:

*Евстигнеев Дмитрий*

Группа: *R3242*

Преподаватель:

*Тропченко Андрей Александрович*

Санкт-Петербург

2021

**Цель:** написать программу для решения задачи №2025 на сайте Timus Online

**Задача:**

2025. Стенка на стенку

*Ограничение времени: 1.0 секунды  
Ограничение памяти: 64 МБ*

Бокс, каратэ, самбо… Классические боевые единоборства пресытили аудиторию. Поэтому известный спортивный канал запускает новый формат соревнований, основанный на традиционной русской забаве — боях стенка на стенку. В соревновании могут участвовать от двух до *k* команд, каждая из которых будет соперничать с остальными. Всего в соревновании примут участие *n* бойцов. Перед началом боя они должны разделиться на команды, каждый боец должен войти ровно в одну команду. За время боя два бойца сразятся, если они состоят в разных командах. Организаторы считают, что популярность соревнований будет тем выше, чем больше будет количество схваток между бойцами. Помогите распределить бойцов по командам так, чтобы максимизировать количество схваток между бойцами, и выведите это количество.

*Исходные данные*

В первой строке дано количество тестов *T* (1 ≤ *T* ≤ 10). В следующих *T* строках перечислены тесты. В каждой из них записаны целые числа *n* и *k* через пробел (2 ≤ *k* ≤ *n* ≤ 104).

*Результат*

Для каждого теста в отдельной строке выведите одно целое число — ответ на задачу.

*Пример*

|  |  |
| --- | --- |
| исходные данные | результат |
| 3 6 3 5 5 4 2 | 12 10 4 |

**Принято системой (JUDGE\_ID: 231802FR):**



**Решение на языке С++:**

#include <iostream>

#include <algorithm>

using namespace std;

int main()

{

int ccount; // количество тестов

cin >> ccount;

for (int i = 0; i < ccount; ++i) {

int n, k; // кол-во команд и бойцов

cin >> n >> k;

int d = n / k; // обычное распредление

int dplus = n % k; // сколько групп будет побольше

int countplus = d + 1; // кол-во человек в dplus группах

int ans = ((n - countplus \* dplus)\*(n - d) + (dplus \* countplus)\*(n - d - 1)) / 2;

cout << ans << endl;

}

return 0;

} **Суть алгоритма:**

Высчитывается размер обычной группы и количество групп побольше на 1. Дальше на каждой итерации происходит подсчет максимального количества схваток. Считаем схватки между людьми из обычной и увеличенной группы.

**Примеры работы программы:**



