XXI командная олимпиада школьников Санкт-Петербурга по информатике и программированию

27 октября 2013 года

Задача A. «Банк»

Задача A. «Банк»

- Идея задачи — Виталик Аксёнов
- Подготовка тестов — Виталик Аксёнов
- Разбор задачи — Виталик Аксёнов

Постановка задачи

Входные данные:

- Сначала каждому гному нужно провести a_i времени у одного из m сотрудников.
- Известно, что гном приходит в банк во время t_i.
- После сотрудника гному нужно провести b_i времени у главного бухгалтера.
- Есть 2 очереди — одна к сотрудникам и одна к бухгалтеру.
- Первый гном из первой очереди идёт к первому освободившемуся сотруднику.

Нужно для каждого гнома сказать, в какое время он выйдет из банка.

Первая очередь

- Первым делом узнаем для каждого гнома время, в которое он встанет во вторую очередь.
- Будем обрабатывать гномов последовательно, по времени прихода.
- Пусть сейчас обрабатываем гнома под номером i.
- Найдём того сотрудника, у которого пусто пустое или у которого находится гном, с минимальным временем перехода во вторую очередь. Пусть это время равно t.
- Отправим i-го гнома к выбранному сотруднику. А время его перехода во вторую очередь равно $t2_i = \max\{t_i, t\} + a_i$.

Первая очередь

- Первым делом узнаем для каждого гнома время, в которое он встанет во вторую очередь.
- Будем обрабатывать гномов последовательно, по времени прихода.
- Пусть сейчас обрабатываем гнома под номером i.
- Найдём того сотрудника, у которого пусто пустое или у которого находится гном, с минимальным временем перехода во вторую очередь. Пусть это время равно t.
- Отправим i-го гнома к выбранному сотруднику. А время его перехода во вторую очередь равно $t2_i = \max\{t_i, t\} + a_i$.

Первая очередь

- Первым делом узнаем для каждого гнома время, в которое он встанет во вторую очередь.
- Будем обрабатывать гномов последовательно, по времени прихода.
- Пусть сейчас обрабатываем гнома под номером i.
- Найдём того сотрудника, у которого пусто пустое или у которого находится гном, с минимальным временем перехода во вторую очередь. Пусть это время равно t.
- Отправим i-го гнома к выбранному сотруднику. А время его перехода во вторую очередь равно $t2_i = \max\{t_i, t\} + a_i$.

Первая очередь

- Первым делом узнаем для каждого гнома время, в которое он встанет во вторую очередь.
- Будем обрабатывать гномов последовательно, по времени прихода.
- Пусть сейчас обрабатываем гнома под номером i.
- Найдём того сотрудника, у которого пусто пустое или у которого находится гном, с минимальным временем перехода во вторую очередь. Пусть это время равно t.
- Отправим i-го гнома к выбранному сотруднику. А время его перехода во вторую очередь равно $t2_i = \max\{t_i, t\} + a_i$.

Вторая очередь

- Отсортируем гномов по времени прихода во вторую очередь. Чтобы не потерять входные номера гномов нам нужно их отсортировать «вместе» с гномами.
- Аналогично первой очереди обрабатываем гномов последовательно.
- Возьмём i-го гнома из очереди, а его номер будет равен ind_i.
- $$\begin{cases} ans_{ind_i} = \max\{ans_{ind_{i-1}}, t2_{ind_i}\} + b_{ind_i}, & i \geq 2 \\ ans_{ind_i} = t2_{ind_i} + b_{ind_i}, & i = 1 \end{cases}.$$

Вторая очередь

- Отсортируем гномов по времени прихода во вторую очередь. Чтобы не потерять входные номера гномов нам нужно их отсортировать «вместе» с гномами.
- Аналогично первой очереди обрабатываем гномов последовательно.
- Возьмём i-го гнома из очереди, а его номер будет равен ind_i.
- $$\begin{cases} ans_{ind_i} = \max\{ans_{ind_{i-1}}, t2_{ind_i}\} + b_{ind_i}, & i \geq 2 \\ ans_{ind_i} = t2_{ind_i} + b_{ind_i}, & i = 1 \end{cases}.$$

Вторая очередь

- Отсортируем гномов по времени прихода во вторую очередь. Чтобы не потерять входные номера гномов нам нужно их отсортировать «вместе» с гномами.
- Аналогично первой очереди обрабатываем гномов последовательно.
- Возьмём i-го гнома из очереди, а его номер будет равен ind_i.
- $$\begin{cases} ans_{ind_i} = \max\{ans_{ind_{i-1}}, t2_{ind_i}\} + b_{ind_i}, & i \geq 2 \\ ans_{ind_i} = t2_{ind_i} + b_{ind_i}, & i = 1 \end{cases}.$$

Задача B. «Ставка»

Задача B. «Ставка»

- Идея задачи — Андрей Станкевич
- Подготовка тестов — Николай Ведерников
- Разбор задачи — Виталик Аксёнов

Постановка задачи

- Дано t тестов.
- Каждый тест состоит из 3 чисел r, g и b — количество красных, зелёных и синих фишек.
- Известно, что мы можем увеличить одно число на 1.
- Нужно найти какое число увеличить, чтобы максимизировать функцию $A \cdot (r^2 + g^2 + b^2) + C \cdot \min\{r, g, b\}$.

Решение

- Переберём тройки $(r+1, g, b)$, $(r, g+1, b)$ и $(r, g, b+1)$.
- Выберем среди них тройку, которая соответствует максимальному числу.
- Выведем ответ, который соответствует выбранной тройке.

Решение

- Переберём тройки $(r+1, g, b)$, $(r, g+1, b)$ и $(r, g, b+1)$.
- Выберем среди них тройку, которая соответствует максимальному числу.
- Выведем ответ, который соответствует выбранной тройке.

Решение

- Переберём тройки $(r+1, g, b)$, $(r, g+1, b)$ и $(r, g, b+1)$.
- Выберем среди них тройку, которая соответствует максимальному числу.
- Выведем ответ, который соответствует выбранной тройке.

Задача C. «Взлом шифра»

Задача C. «Взлом шифра»

- Идея задачи — Артём Васильев
- Подготовка тестов — Артём Васильев
- Разбор задачи — Виталик Аксёнов

Постановка задачи

Нужно выписать последовательность чисел от 1 до n длиной не больше $2n!$, в которой будет встречаться любая перестановка чисел от 1 до n, как подстрока.

Генерация

- Сгенерируем все перестановки чисел от 2 до n.
- Будем брать сгенерированные перестановки последовательно.
- Пусть взяли перестановку p. Добавим к ответу: $ans = ans + 1 + p + 1 + p$.

Генерация

- Сгенерируем все перестановки чисел от 2 до n.
- Будем брать сгенерированные перестановки последовательно.
- Пусть взяли перестановку p. Добавим к ответу: $ans = ans + 1 + p + 1 + p$.

Объяснение

- Пусть s — любая перестановка от 1 до n.
- Пусть t — перестановка с 1 на первом месте, получившаяся с помощью циклического сдвига s.
- Тогда перестановка s была добавлена на шаге добавления перестановки t к ответу.
- Каждая добавленная перестановка p добавила к длине ответа $2n$.
- Тогда длина ответа равна $2n \cdot (n-1)! = 2n!$.

Объяснение

- Пусть s — любая перестановка от 1 до n.
- Пусть t — перестановка с 1 на первом месте, получившаяся с помощью циклического сдвига s.
- Тогда перестановка s была добавлена на шаге добавления перестановки t к ответу.
- Каждая добавленная перестановка p добавила к длине ответа $2n$.
- Тогда длина ответа равна $2n \cdot (n-1)! = 2n!$.

Задача D. «Болезнь»

Задача D. «Болезнь»

- Идея задачи — Анна Малова
- Подготовка тестов — Анна Малова
- Разбор задачи — Анна Малова

Постановка задачи

Входные данные:

- Есть n бактерий — возбудителей болезни и m различных тестов.
- Каждый тест проверяет наличие или отсутствие некоторых видов бактерий.
- Тест выдает положительный результат, если найден хотя бы один возбудитель.

Необходимо выяснить про каждую бактерию, заражен ли ею человек. Дополнительное условие: данные тестов могут содержать ошибку.

Общие соображения

- Если результат теста отрицательный, то человек не может быть заражен ни одним из возбудителей, указанных в тесте.
- Если тест проверяет ровно одну бактерию и результат теста положителен, данный возбудитель точно является причиной болезни.

Общие соображения

- Если результат теста отрицательный, то человек не может быть заражен ни одним из возбудителей, указанных в тесте.
- Если тест проверяет ровно одну бактерию и результат теста положителен, данный возбудитель точно является причиной болезни.

Решение

- Найдем все бактерии, которые точно не являются причиной болезни.
- Выкинем из тестов с положительным результатом проверку на этих возбудителей.
- Если в тесте осталась ровно одна бактерия — она точно является причиной заболевания
- Если тест больше не проверяет ни одну бактерию, входные данные противоречивы.

Решение

- Найдем все бактерии, которые точно не являются причиной болезни.
- Выкинем из тестов с положительным результатом проверку на этих возбудителей.
- Если в тесте осталась ровно одна бактерия — она точно является причиной заболевания
- Если тест больше не проверяет ни одну бактерию, входные данные противоречивы.

Решение

- Найдем все бактерии, которые точно не являются причиной болезни.
- Выкинем из тестов с положительным результатом проверку на этих возбудителей.
- Если в тесте осталась ровно одна бактерия — она точно является причиной заболевания
- Если тест больше не проверяет ни одну бактерию, входные данные противоречивы.

Задача E. «Разделение королевства»

Задача E.
«Разделение королевства»

- Идея задачи — Анна Малова
- Подготовка тестов — Виталий Демьянюк
- Разбор задачи — Анна Малова

Постановка задачи

Входные данные:
В королевстве n замков, i-ый замок находится в точке $(x_i + 0.5, y_i + 0.5)$. Необходимо разделить королевство так, чтобы выполнялись следующие условия:

- После разделения любые два замка должны находится в разных частях королевства.
- Разделение производится прямыми, параллельными осям координат.
- Для горизонтальных прямых x, а для вертикальных y, координата всех точек на оси должна быть целым числом.
- Количество прямых не должно превышать $n - 1$.

Решение

- Отсортируем замки по x координате, между каждой парой соседних замков проведем прямую.
- Посмотрим на замки, которые имеют одинаковую координату по x.
  Для каждого такого x отсортируем замки по y координате. Проведем прямую между каждой парой соседних замков.
- Заметим, что при таком разделении, количество прямых не превышает $n - 1$.

Решение

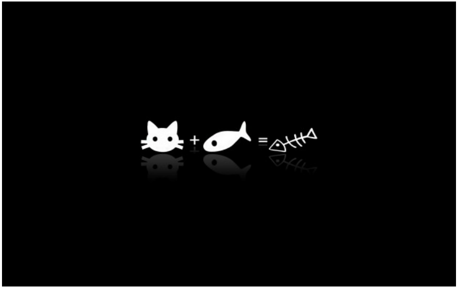
- Отсортируем замки по x координате, между каждой парой соседних замков проведем прямую.
- Посмотрим на замки, которые имеют одинаковую координату по x.
  Для каждого такого x отсортируем замки по y координате. Проведем прямую между каждой парой соседних замков.
- Заметим, что при таком разделении, количество прямых не превышает $n - 1$.

Решение

- Отсортируем замки по x координате, между каждой парой соседних замков проведем прямую.
- Посмотрим на замки, которые имеют одинаковую координату по x.
  Для каждого такого x отсортируем замки по y координате. Проведем прямую между каждой парой соседних замков.
- Заметим, что при таком разделении, количество прямых не превышает $n - 1$.

Задача F. «Загадочное уравнение»

Задача F.
«Загадочное уравнение»



- Идея задачи — Никита Иоффе
- Подготовка тестов — Никита Иоффе
- Разбор задачи — Демид Кучеренко

Постановка задачи

Дано уравнение $x + y + xy = n$. Найти количество пар целых неотрицательных чисел x и y, которые являются решениями этого уравнения.

Перебор всех решений

- Заметим, что многочлен $x + y + xy$ симметричен относительно своих переменных.
- Это значит, что если какая то пара $x = a$ и $y = b$ является решением уравнения, то пара $x = b$ и $y = a$ также является решением данного уравнения.
- Тогда, подобно алгоритму проверки числа на простоту, переберем все x не превосходящие $\sqrt{n}$.
- Для каждого x восстановим y и, если оно целое, добавим пары (x, y) и (y, x) в ответ.
- Не забудем, что если $x = y$, то нужно добавить только одну пару.

Перебор всех решений

- Заметим, что многочлен $x + y + xy$ симметричен относительно своих переменных.
- Это значит, что если какая то пара $x = a$ и $y = b$ является решением уравнения, то пара $x = b$ и $y = a$ также является решением данного уравнения.
- Тогда, подобно алгоритму проверки числа на простоту, переберем все x не превосходящие $\sqrt{n}$.
- Для каждого x восстановим y и, если оно целое, добавим пары (x, y) и (y, x) в ответ.
- Не забудем, что если $x = y$, то нужно добавить только одну пару.

Раскладываем на множители

- $n = x + y + xy \Rightarrow n + 1 = 1 + x + y + xy = (x + 1)(y + 1)$.
- Нужно просто разложить $n + 1$ на множители.

Задача G. «Осенний парк»

Задача G. «Осенний парк»

- Идея задачи — Георгий Корнеев
- Подготовка тестов — Олег Давыдов
- Разбор задачи — Виталик Аксёнов

Постановка задачи

Дано поле $n \times m$ с препятствиями. Нужно вычислить количество путей из одной точки в другую, которые отличаются по длине от кратчайшего на два.

Нахождение количества кратчайших путей

- Запустим BFS из стартовой точки. Пусть ответ будет храниться в массиве cnt.
- А длины кратчайших путей будут в массиве d.
- Пусть вытащили на очередном шаге вершину v.
- Мы пробегаем по всем её соседям. Пусть рассматриваем сейчас соседа u.
- Если расстояние до u больше чем расстояние до v, то $cnt[u] = (cnt[u] + cnt[v]) \mod 10^9 + 9$.

Нахождение количества кратчайших путей

- Запустим BFS из стартовой точки. Пусть ответ будет храниться в массиве cnt.
- А длины кратчайших путей будут в массиве d.
- Пусть вытащили на очередном шаге вершину v.
- Мы пробегаем по всем её соседям. Пусть рассматриваем сейчас соседа u.
- Если расстояние до u больше чем расстояние до v, то $cnt[u] = (cnt[u] + cnt[v]) \mod 10^9 + 9$.

Нахождение количества кратчайших путей

- Запустим BFS из стартовой точки. Пусть ответ будет храниться в массиве cnt.
- А длины кратчайших путей будут в массиве d.
- Пусть вытащили на очередном шаге вершину v.
- Мы пробегаем по всем её соседям. Пусть рассматриваем сейчас соседа u.
- Если расстояние до u больше чем расстояние до v, то $cnt[u] = (cnt[u] + cnt[v]) \mod 10^9 + 9$.

Подправление алгоритма

- Добавляем для вершины v вершину v', в которой будут заканчиваться пути длиннее кратчайшего на 2.
- Добавим ребро из v в u', если было ребро из v в u и $d[v] = d[u] + 1$.
- Таким образом, применяя алгоритм для количества кратчайших путей, мы находим ответ.

Подправление алгоритма

- Добавляем для вершины v вершину v', в которой будут заканчиваться пути длиннее кратчайшего на 2.
- Добавим ребро из v в u', если было ребро из v в u и $d[v] = d[u] + 1$.
- Таким образом, применяя алгоритм для количества кратчайших путей, мы находим ответ.

Подправление алгоритма

- Добавляем для вершины v вершину v', в которой будут заканчиваться пути длиннее кратчайшего на 2.
- Добавим ребро из v в u', если было ребро из v в u и $d[v] = d[u] + 1$.
- Таким образом, применяя алгоритм для количества кратчайших путей, мы находим ответ.

Задача H. «Палиндромные числа»

Задача H. «Палиндромные числа»

- Идея задачи — Илья Малиновский
- Подготовка тестов — Павел Кунявский
- Разбор задачи — Павел Кунявский

Постановка задачи

Определения:

- Число называется палиндромным, если читается одинаково слева направо и справа налево.
- Число x называется межпалиндромным, если $x+1$ и $x-1$ — палиндромные.

Задача:

- Найти количество межпалиндромных чисел в отрезке $[L,\ R]$.

Решение: общие соображения

- Будем считать, что чисел не очень много. Найдем их все, после этого задача решается любым поиском, потому что таких чисел немного.
- Посмотрим на последнюю цифру межпалиндромного числа x. Если она не 0 и не 9, то ни при сложении, ни при вычитании никаких переносов не происходит. Тогда и у $x+1$ и $x-1$ последняя ненулевая цифра, совпадает с первой цифрой x.

Решение: общие соображения

- Будем считать, что чисел не очень много. Найдем их все, после этого задача решается любым поиском, потому что таких чисел немного.
- Посмотрим на последнюю цифру межпалиндромного числа x. Если она не 0 и не 9, то ни при сложении, ни при вычитании никаких переносов не происходит. Тогда и у $x+1$ и $x-1$ последняя ненулевая цифра, совпадает с первой цифрой x.

Решение: общие соображения

- Это бывает в двух случаях: либо x — однозначное, либо последняя цифра 1.
- Однозначные сразу добавим в ответ. После этого можно считать, что число заканчивается на 0, 1 или 9.

Решение: числа, заканчивающиеся на 0

- Так как $x + 1$ палиндромное, то первая цифра числа равна 1.
- Так как $x - 1$ палиндромное, то первая цифра $x - 1$ равна 9.
- Это возможно, только если все остальные цифры нули, то есть $x = 10^k$. Все такие числа подходят.

Решение: числа, заканчивающиеся на 1 или 9

- Пусть x заканчивается на 1.
- Так как $x + 1$ палиндромное, то первая цифра числа равна 2.
- Применяя по очереди условия на палиндромность $x - 1$ и $x + 1$, для каждой цифры можно доказать, что она равна или первой двойке, или последней единице, или одному из нулей, идущих сразу перед 1, если такие есть.

Решение: числа, заканчивающиеся на 1 или 9

- Пусть x заканчивается на 1.
- Так как $x + 1$ палиндромное, то первая цифра числа равна 2.
- Применяя по очереди условия на палиндромность $x - 1$ и $x + 1$, для каждой цифры можно доказать, что она равна или первой двойке, или последней единице, или одному из нулей, идущих сразу перед 1, если такие есть.

Решение: числа, заканчивающиеся на 1 или 9

- Таким образом межпалиндромное число, заканчивающееся на 1, состоит из цифр 0,1,2.
- Аналогично, межпалиндромное число, заканчивающееся на 9, состоит из цифр 7,8,9.

Решение: перебор

- Переберём последние 9 цифр числа. Есть $3^9 = 19683$ вариантов.
- Либо по $x + 1$, либо по $x - 1$ восстановим все число.
- Проверим что оно подошло.
- Всего подходит 131 число.

Задача I. «Необычный экспонат»

Задача I. «Необычный экспонат»

- Идея задачи — Артур Рязанов
- Подготовка тестов — Демид Кучеренко
- Разбор задачи — Виталик Аксёнов, Андрей Комаров

Постановка задачи

- Даны количества инверсий на всех подотрезках длины k перестановки чисел от 1 до n.
- Нужно восстановить любую перестановку, удовлетворяющую условию.
- Гарантировалось, что всегда такая перестановка существует.

План решения

- Переберём, <u>форму</u> какой перестановки имеют первые $k-1$ число ответа.
  - Например, последовательность $\langle 100, 500, 239, 17 \rangle$ <u>имеет форму</u> $\langle 2, 4, 3, 1 \rangle$.
- Попробуем построить подходящий под эту форму ответ.

Построения ответа по форме

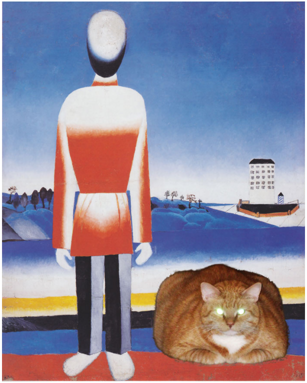
- Пускай что-то известно про первые m чисел (изначально, $m = k - 1$).
- Выясним что-нибудь про очередное число.
- Будем строить граф G. В G есть ребро $i \longrightarrow j$, если известно, что в строящемся массиве $a_i < a_j$.
- Про каждую пару из $\{a_{m-k+1}, a_{m-k+2}, \ldots, a_m\}$ известно, какой элемент из пары максимальный.
- Перебрав k вариантов того, как будет соотноситься с ними a_{m+1}, посчитаем количество инверсий на отрезке $a_{m-k+2..m+1}$ в каждом из этих случаев.

Восстановление ответа

- Получим нужное число инверсий не более, чем в одном из случаев.
  - Не получили ни в одном — переходим к следующей форме.
  - Получили одно — переходим к следующему m.
- По условию задачи хотя бы для одной формы дойдём до конца.
- Запустим алгоритм топологической сортировки на получившемся графе.
- Полученная перестановка и будет ответом.

Задача J. «Супрематизм»

Задача J. «Супрематизм»

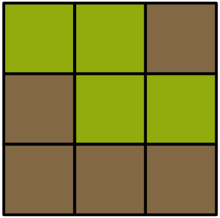


- Идея задачи — Анна Малова
- Подготовка тестов — Андрей Комаров
- Разбор задачи — Андрей Комаров

Постановка задачи

Входные данные:

- Таблица из $n \times m$ чисел (<u>цветов</u> клеток).
- Можно перекрашивать строку в цвет c, если в ней более, чем $\left\lfloor \frac{m}{2} \right\rfloor$ клеток цвета c.
- Аналогично, столбец — если в нём больше $\left\lfloor \frac{n}{2} \right\rfloor$ клеток цвета c.



Нужно раскрасить всю таблицу в один цвет или сказать, что это сделать нельзя.

Важное наблюдение

Если ответ есть, то есть ответ, в котором все перекрашивания происходят в один цвет.

- Допустим, что это не так.
- В конце таблица полностью раскрашена в цвет c.
- Индукция по числу r раскрашиваний в цвета, отличные от c:
    - $r = 0$ — ничего делать не надо.
    - $r > 0$. Последнее перекрашивание в цвет $c' \neq c$. Если бы его не было, всё равно можно было бы докрасить.

Важное наблюдение

Если ответ есть, то есть ответ, в котором все перекрашивания происходят в один цвет.

- Допустим, что это не так.
- В конце таблица полностью раскрашена в цвет c.
- Индукция по числу r раскрашиваний в цвета, отличные от c:
  - $r = 0$ — ничего делать не надо.
  - $r > 0$. Последнее перекрашивание в цвет $c' \neq c$. Если бы его не было, всё равно можно было бы докрасить.

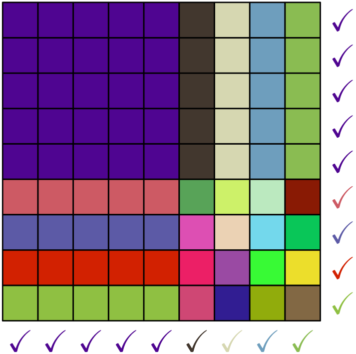
Важное наблюдение

Если ответ есть, то есть ответ, в котором все перекрашивания происходят в один цвет.

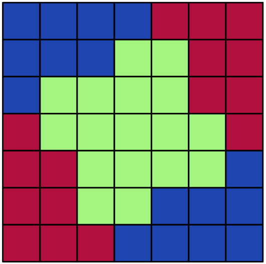
- Допустим, что это не так.
- В конце таблица полностью раскрашена в цвет c.
- Индукция по числу r раскрашиваний в цвета, отличные от c:
    - $r = 0$ — ничего делать не надо.
    - $r > 0$. Последнее перекрашивание в цвет $c' \neq c$. Если бы его не было, всё равно можно было бы докрасить.

Цвета-кандидаты

- Если в цвет в начальной таблице нельзя раскрасить ни один столбец и ни одну строку, то и всю таблицу в него нельзя раскрасить.
- Цвета-кандидаты — цвета, в которые можно раскрасить хотя бы одну строку или столбец.
    - Их не более $n + m$.



Почему не хватит одного цвета?



- Рассматривать только цвет, которого больше всего, нельзя (WA12).
- На этой картинке больше всего центрального цвета, однако, в него раскрасить нельзя.
- А в два других можно.

Решение

- Переберём всех кандидатов.
- Проверим, можно ли раскрасить в этот цвет c всё поле.
    - Для каждой строки и каждого столбца будем считать сколько там элементов цвета c.
    - Пробежимся по всем строкам, если в ней от $\lfloor \frac{m}{2} \rfloor + 1$ до $m - 1$ элементов цвета c, то пробежимся по ней, перекрашивая не-c в c и пересчитывая количества.
    - Аналогично для столбцов.

Решение

- Переберём всех кандидатов.
- Проверим, можно ли раскрасить в этот цвет c всё поле.
  - Для каждой строки и каждого столбца будем считать сколько там элементов цвета c.
  - Пробежимся по всем строкам, если в ней от $\left\lfloor \frac{m}{2} \right\rfloor + 1$ до $m - 1$ элементов цвета c, то пробежимся по ней, перекрашивая не-c в c и пересчитывая количества.
  - Аналогично для столбцов.

Решение

- Переберём всех кандидатов.
- Проверим, можно ли раскрасить в этот цвет c всё поле.
  - Для каждой строки и каждого столбца будем считать сколько там элементов цвета c.
  - Пробежимся по всем строкам, если в ней от $\left\lfloor \frac{m}{2} \right\rfloor + 1$ до $m - 1$ элементов цвета c, то пробежимся по ней, перекрашивая не-c в c и пересчитывая количества.
  - Аналогично для столбцов.

Сложность

- Каждая строка и каждый столбец перекрашиваются не более одного раза.
- Перекрашивание происходит за $O(n)$ или $O(m)$.
- Нахождение очередной цели для перекрашивания — $O(n+m)$.
- Итого, проверка того, можно ли покрасить в данный цвет — $O((n+m)^2)$.
- Всего цветов для проверки — $n+m$. Итоговая сложность — $O((n+m)^3)$.

Вопросы?