XXV командный чемпионат школьников Санкт-Петербурга по программированию

5 ноября 2017 года

Жюри соревнований

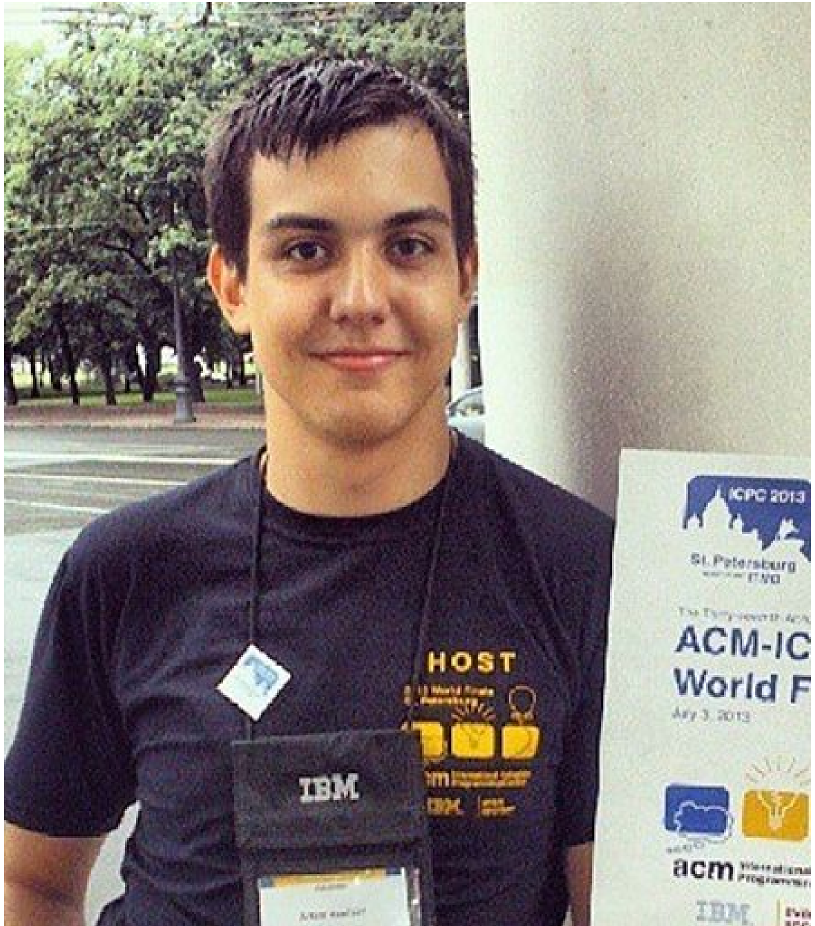
Андрей Станкевич

Георгий Корнеев

Олег Давыдов

Николай Будин

Артем Васильев

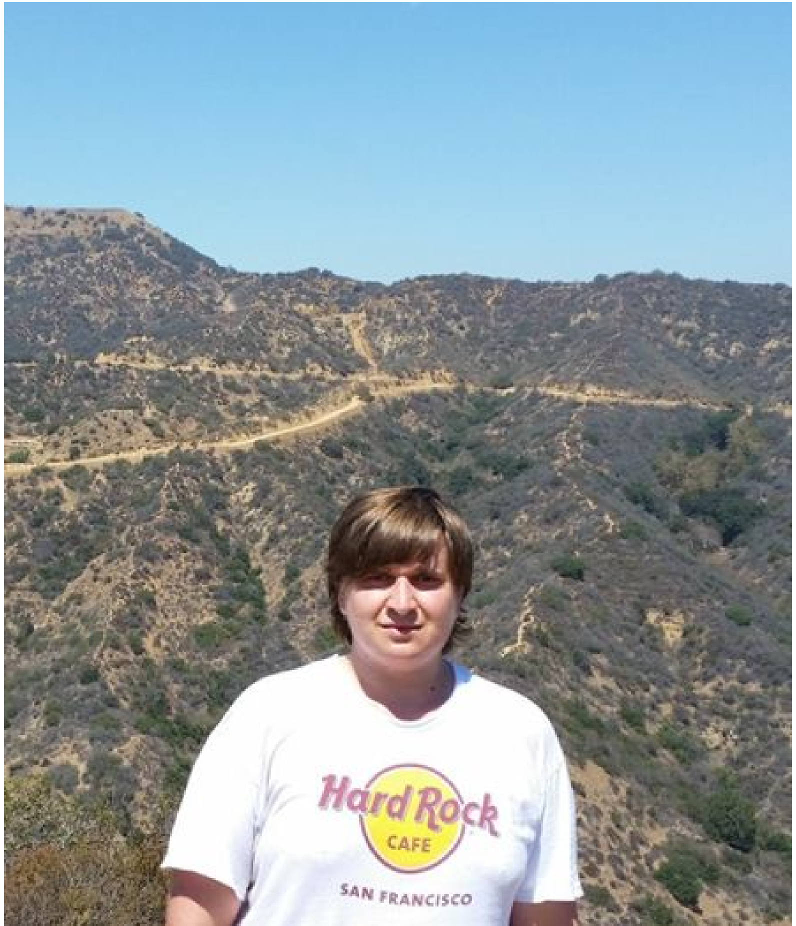


Николай Ведерников

Антон Гардер

Михаил Дворкин

Андрей Комаров



Сергей Копелиович

Геннадий Короткевич

Евгений Курпилянский

Павел Кунявский

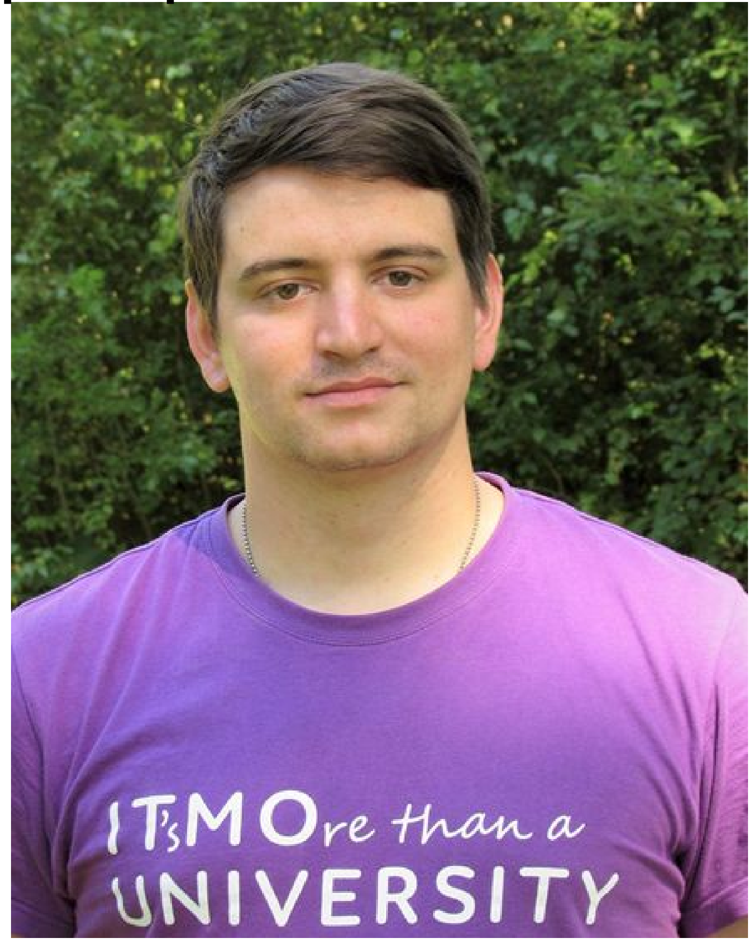
Нияз Нигматуллин

Борис Минаев

Дмитрий Филиппов

Михаил Путилин

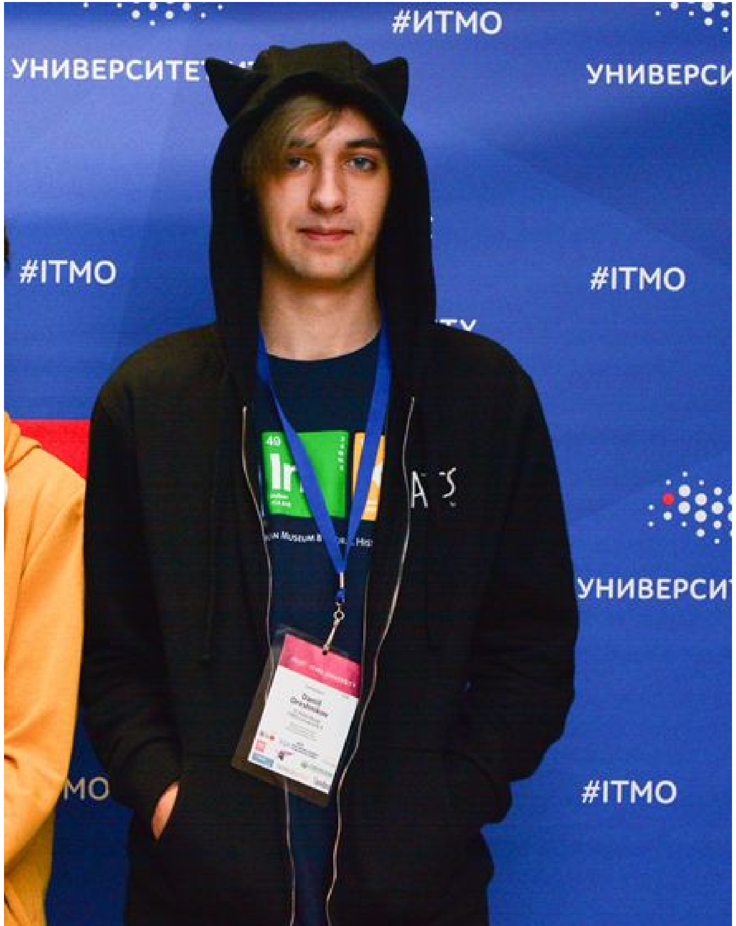
Григорий Шовкопляс



Александра Дроздова

Арсений Кириллов

Даниил Орешников

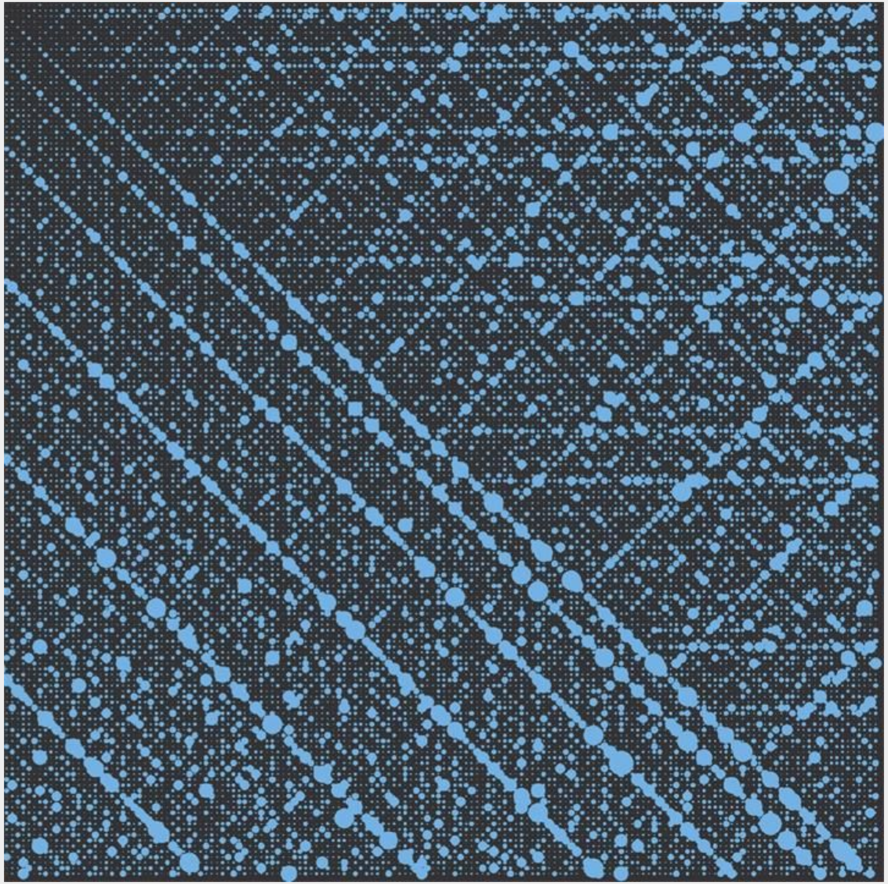


Дмитрий Саютин

Михаил Ютман

Задача A
Закономерности

Автор задачи:
Андрей Станкевич
Разработка задачи:
Андрей Станкевич



Постановка задачи

- Поле размера $n \times n$, в нём числа от **1** до n^2, подряд.
- Выделить из них те, которые имеют не более k делителей.
- Нарисовать картинку из «`.`» и «`*`».

Решение

Решение

- Для каждого числа находим количество делителей и сравниваем с ***k***.
- Количество делителей можно искать как-нибудь быстро (например, аналогом решета Эратосфена) или хотя бы за sqrt(***x***), где ***x*** — очередное число.
- А можно и просто перебором всех возможных от **1** до ***x*** с проверкой, а не делитель ли это.

Задача B
Интересная экскурсия

Автор задачи:
Артем Васильев
Разработка задачи:
Артем Васильев

Постановка задачи

- Дан ориентированный граф, рёбра раскрашены в какие-то цвета.
- Найти цикл, в котором не встречается двух одинаковых цветов подряд.
- Можно проходить дважды через одну и ту же вершину.

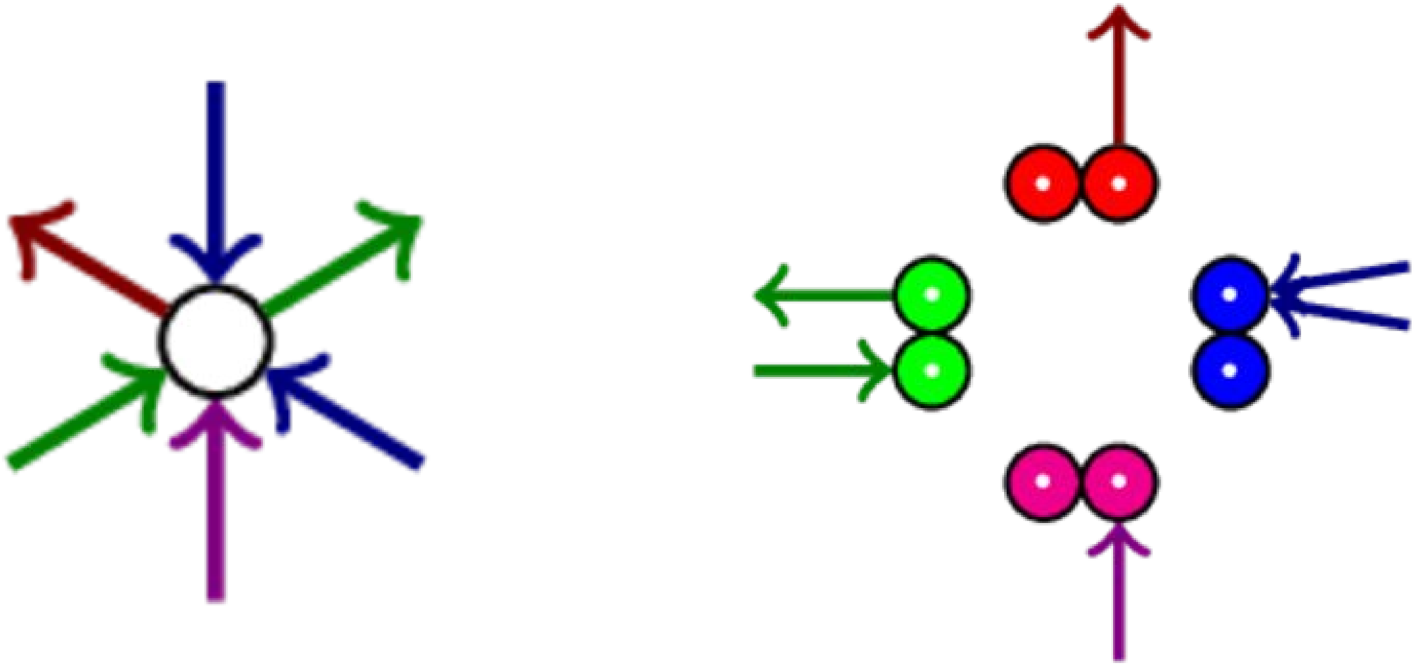
Решение

- Построим вспомогательный граф, любой цикл в котором даст нам правильный ответ на исходную задачу.
- Найдём цикл во вспомогательном графе (например, поиском в глубину с тремя цветами вершин).
- По найденному циклу восстановим ответ.

Решение

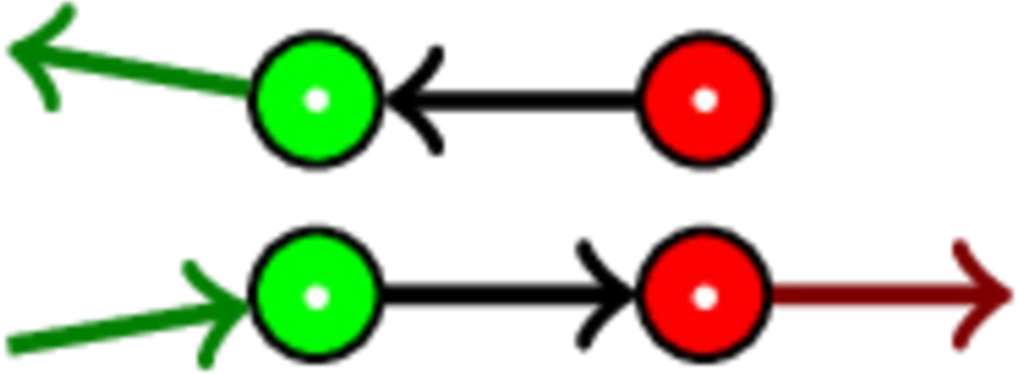
- Что это за вспомогательный граф?
- Разделим каждую вершину на ***2k*** новых, где ***k*** — количество различных цветов на инцидентных рёбрах.
- Нас интересуют и входящие, и исходящие.
- Входящие рёбра направим в чётные вершины, исходящие — в нечётные, каждая пара вершин будет соответствовать своему цвету.

Решение



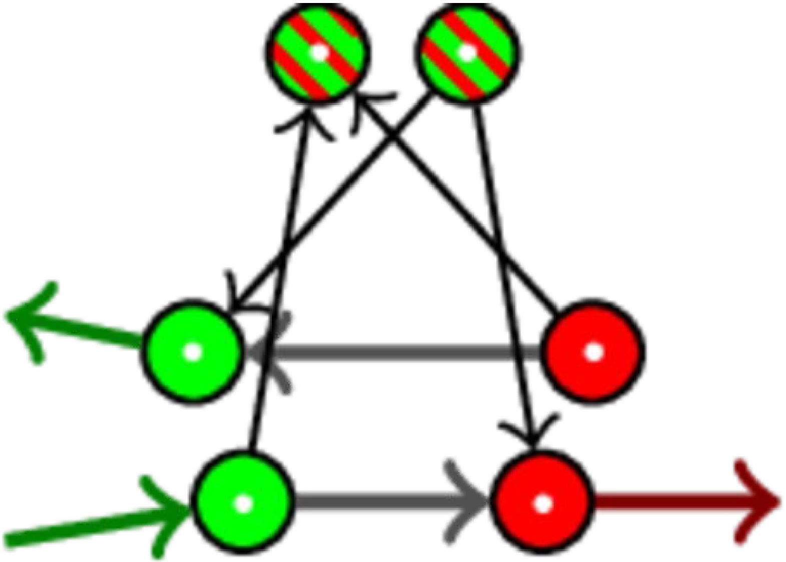
Решение

- Возьмём первые два цвета и соединим соответствующие вершины (от «входящей» одного цвета ребро к «исходящей» другого).



Решение

- Построим ещё две вспомогательные вершины, для «объединённого» цвета.
- Добавим нужные рёбра.



Решение

- Повторяя предыдущие шаги, объединим все цвета у каждой вершины.
- Вспомогательный граф готов!
- Во вспомогательном графе будет довольно много как рёбер, так и вершин, но асимптотически он не больше, чем исходный.
- Грубая оценка: не более $\boldsymbol{8m}$ вершин для графа с $\boldsymbol{m}$ рёбрами.

Решение

- Требуется восстановить ответ.
- При копировании рёбер во вспомогательный граф сохраним их номера.
- Вспомогательным рёбрам в качестве номера запишем ноль.
- Из найденного цикла выкинем при выводе все нулевые рёбра.

Задача C
Прыжки с поворотом

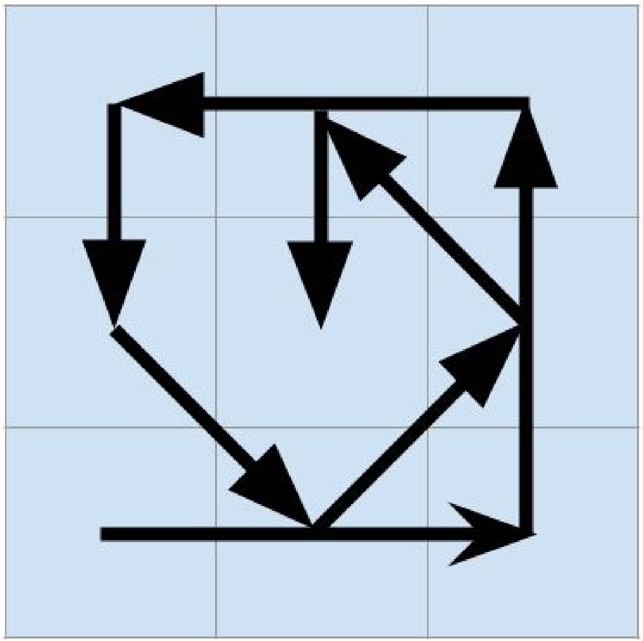
Автор задачи:
Артем Васильев
Разработка задачи:
Николай Будин

Постановка задачи

- Обойти поле размера $n \times m$.
- Каждый следующий переход по направлению должен лежать строго слева предыдущего (поворот против часовой стрелки).
- Можно прыгать «далеко», при этом промежуточные клетки не считаются посещёнными.

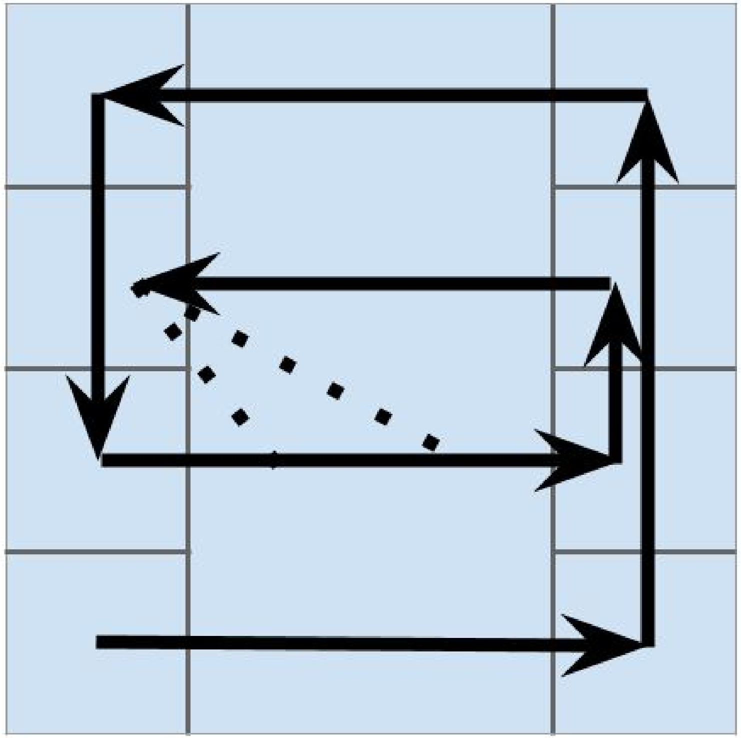
Решение

- Общая идея: конструктив и небольшой разбор случаев.
- Для полей **1×1**, **1×2** и **2×1** ответ тривиален.
- Для поля **1×n** и **n×1** (где ***n*** > 2) ответ «`No`».
- Для поля **3×3** возможный ответ показан справа.



Решение (продолжение)

- Поле ***n*×*m*** можно свести к полю **(*n* − 2)×*m*** или ***n*× (*m* − 2)**
- Так любое оставшееся поле можно привести к ***n*×0**, **0×*n*** (и тогда задача решена) или к **3×3** (который разобран выше)



Возможные проблемы

- Куски решения (сведения, маленькие случаи) нужно аккуратно «склеивать» между собой

Задача D
Подсчеты в строю

Автор задачи:
Дмитрий Филиппов
Разработка задачи:
Дмитрий Филиппов

Постановка задачи

- В строю стоит n солдат, каждый смотрит либо налево, либо направо, у каждого есть рост h_i.
- Солдат i видит солдата j, если между ними нет солдат с ростом больше h_j.
- Для каждого солдата найти, сколько других солдат он видит.

Решение

- Найдем ответ для всех солдат, которые смотрят налево (для остальных – аналогично).
- Пусть $\boldsymbol{i}$-й солдат видит солдат с номерами $\boldsymbol{j_1} < \boldsymbol{j_2} < \ldots < \boldsymbol{j_k}$
- Как изменится список при переходе к солдату $\boldsymbol{i} + 1$?
    - В список добавляется солдат $\boldsymbol{i}$.
    - Некоторых (низких) солдат из старого списка солдат $\boldsymbol{i} + 1$ не видит.

Решение

- Наблюдение: $\boldsymbol{h_{j1}} \geq \boldsymbol{h_{j2}} \geq \ldots \geq \boldsymbol{h_{jk}}$.
- Следовательно, солдаты, которых больше не видит ($\boldsymbol{i}$ + *1*)-й, — в конце списка.
- Используем стек для хранения списка солдатов, которых видит текущий солдат.
- При переходе к ($\boldsymbol{i}$ + *1*)-му солдату, удаляем несколько солдат из стека, а затем добавляем $\boldsymbol{i}$-го солдата.
- Асимптотика: $\boldsymbol{O(n)}$.

Задача E
Разные цифры

Автор задачи:
Андрей Станкевич
Разработка задачи:
Нияз Нигматуллин

Постановка задачи

- Дано ***n***.
- Найти ближайшее сверху натуральное число без пары одинаковых цифр подряд.

Решение

- Найдём разряд, в котором будет первое отличие:

$$\begin{array}{cccccccl} a_1 & a_2 & \dots & \boldsymbol{a_i} & \dots & a_n & & \text{(изначальное число)} \\ = & = & = & \boldsymbol{\neq} & & & & \\ b_1 & b_2 & \dots & \boldsymbol{b_i} & \dots & b_n & & \text{(итоговое число)} \end{array}$$

  - Слева от него нет пары одинаковых цифр.
  - Можно увеличить $\boldsymbol{a_i}$ на 1 (а если $\boldsymbol{a_i}+1=\boldsymbol{a_{i-1}}$, то на 2).
  - Из таких разрядов выберем самый правый.

Решение

- В этом разряде сделаем увеличение на 1 или 2:
  $a_1\ a_2\ \ldots\ \boldsymbol{a_i}\ \ldots$ (изначальное число)

  $a_1\ a_2\ \ldots\ \boldsymbol{a_i+\{1,2\}}\ 0\ 1\ 0\ldots$ (итоговое число)
- Остальные разряды заполним 0 и 1 по очереди, начиная с 0.

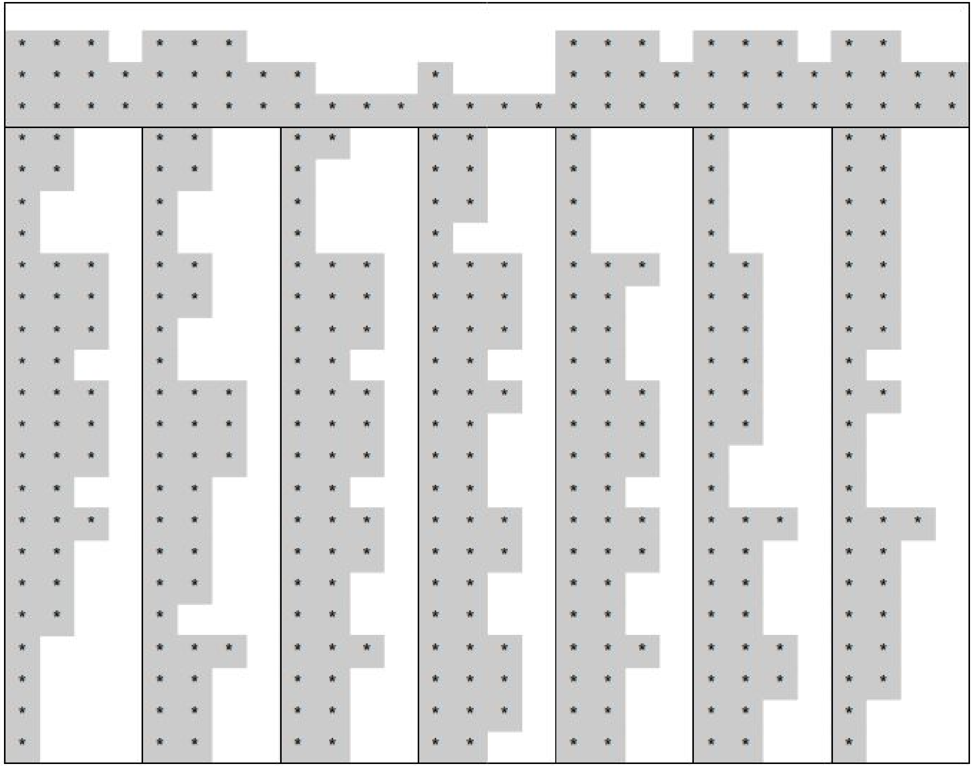
Задача F
Рисование

Автор задачи:
Михаил Дворкин
Разработка задачи:
Борис Минаев

Постановка задачи

- Лист клеточной бумаги разбили на квадратаы 4×4, в каждом квадрате записали число от 4 до 12.
- Необходимо закрасить в каждом квадрате ровно столько клеток, какое число в нем записано.
- Закрашенная область должна быть связна.
- Незакрашенная область должна быть связна.
- Первая и последняя клетка каждой строки и каждого столбца считаются соседними.

Решение



Задача G
Последняя битва

Автор задачи:
Нияз Нигматуллин
Разработка задачи:
Александра Дроздова

Постановка задачи

- Дана перестановка чисел $\boldsymbol{a_1}$, $\boldsymbol{a_2}$, …, $\boldsymbol{a_n}$.
- Нужно найти минимальное число позиций, на которое надо циклически сдвинуть перестановку влево так, чтобы чтобы в ней не оказалось позиции ***i***, число на которой равно ***i***.

Решение

- Заметим, что ответ, если он есть, не более $\boldsymbol{n} - 1$.
- Также для каждой позиции $\boldsymbol{i}$ есть ровно один циклический сдвиг перестановки $\boldsymbol{a_1}, \ldots, \boldsymbol{a_n}$ на $\boldsymbol{k}$ позиций $(0 \le \boldsymbol{k} \le \boldsymbol{n} - 1)$, при котором на этой позиции окажется число $\boldsymbol{i}$. Назовем его плохим.
- Он равен остатку числа $\boldsymbol{i} - \boldsymbol{a_i}$ при делении на $\boldsymbol{n}$.

Решение

- Переберем позицию, узнаем какой циклический сдвиг является плохим для нее.
- Для каждого циклического сдвига на $\boldsymbol{k}$ позиций ($0 \leq \boldsymbol{k} \leq \boldsymbol{n} - 1$) сохраним был ли он плохим для какой-то позиции.
- Переберем циклический сдвиг на $\boldsymbol{k}$ позиций ($0 \leq \boldsymbol{k} \leq \boldsymbol{n} - 1$), если он ни разу не был плохим, то он является ответом.
- Если ни один не подошел, то значит такого циклического сдвига нет, и ответ -1.

Задача H
Расписание

Автор задачи:
Антон Гардер
Разработка задачи:
Арсений Кириллов

Постановка задачи

- Дано расписание уроков в школе и число ***k***: количество часов недосыпа, которое можно накопить.
- Спать можно только от полуночи до первого урока, не более десяти часов.
- Нужно найти первый день, в который ученик проспит свой первый урок, если он хочет спать в среднем по 8 часов в сутки.

Решение

- Для каждого дня нам достаточно знать максимальное число часов, которое ученик может спать в этот день.
- Это число равно минимуму из 10 и времени начала первого урока в этот день минус 1, если в этот день вообще есть уроки.

Решение

- Посчитаем недосып после первой недели и после второй.
- Если в какой-то день первых двух недель случился недосып хотя бы ***k*** часов, то выведем этот день.

Решение

- Если недосып после двух недель не больше, чем после одной недели, то и после каждой недели он будет не больше, чем после первой, а значит ученик никогда не пропустит урок.
- Иначе после каждой недели он будет увеличиваться на разность этих двух чисел.
- Тогда надо перебрать все дни и вывести первый день, в котором недосып будет хотя бы ***k*** часов.

Задача I
Пицца

Автор задачи:
Дмитрий Саютин
Разработка задачи:
Геннадий Короткевич

Постановка задачи

- Есть $\boldsymbol{n}$ ингредиентов для пиццы, $\boldsymbol{m}$ друзей.
- У каждого друга – пожелания, какие ингредиенты добавить, а какие не добавлять.
- Если хотя бы одно пожелание выполнено, друг рад.
- Найти количество способов обрадовать всех друзей.

Решение

- Формула включений-исключений.
- Перебираем подмножество друзей ***S***, ни одно пожелание которых не будет выполнено.
- Узнаём про все ингредиенты в пожеланиях друзей из ***S***.
- Если нет противоречий, добавляем к ответу $\pm 2^{k}$.
    - ***k*** – количество ингредиентов, не участвующих в пожеланиях друзей из подмножества.

Решение

- Сложность решения?
- Перебор масок циклом: $O(2^m (\boldsymbol{n} + \Sigma \boldsymbol{a_i}))$.
    - Неэффективно.
- Рекурсивный перебор: $\boldsymbol{n} + \Sigma\, 2^i \boldsymbol{a_i} = O(\boldsymbol{n} + 2^m \max \boldsymbol{a_i})$.
    - Подразумевалось.
- Альтернативное решение: битовое сжатие, $O(2^m \boldsymbol{n} / 32)$.

Задача J
Ретвинтим twinter

Автор задачи:
Михаил Дворкин
Разработка задачи:
Михаил Дворкин

Постановка задачи

- Дана цепочка «твинтов» вида:
  текст первого твинта (1/***n***)
  …
  текст последнего твинта (***n***/***n***)
- Переформатировать её в как можно более короткую цепочку с новым ограничением длины (280 вместо 140).

Решение

- Переберём длину числа ***m*** (числа твинтов в ответе).
- Жадно наполняем твинты новой цепочки.
    - Оставляем справа место для « (***i***/***m***) ».
    - При этом ***i*** нам известно.
    - А для ***m*** отводим известное нам число символов.
- Количество твинтов ***m*** получилось ровно такой длины?
    - Да → вставляем ***m*** на отведенные места. Готово.
    - Нет → переходим к следующей длине ***m***.

Задача K
Дробление

Автор задачи:
Даниил Орешников
Разработка задачи:
Даниил Орешников

Постановка задачи

- Даны четыре целых положительных числа.
- Найти и вывести такую их перестановку (***a***, ***b***, ***c***, ***d***), чтобы сумма дробей ***a***/***b*** + ***c***/***d*** была минимальна.

Решение 1

- Переберем все перестановки этих чисел.
  - всего 4 числа — можно даже перебрать вручную.
- Для каждой посчитаем значение этого выражения.
- Выберем ту, для которой это значение минимально.
- Замечания:
  - могли быть проблемы с точностью, которые можно было решить, использовав длинные типы данных.

Решение 2

- Посмотрим на перестановку $a \le b \le c \le d$.
- Утверждение: (a, c, b, d) – перестановка, дающая минимальное дробление.
- Действительно, если (x, y, z, t) дает минимальную, то
  - $x \le y$
  - $z \le t$
  - Так как иначе мы поменяем их друг с другом и результат уменьшится.

Решение 2

- В (x, y, z, t) $x \le y$ и $z \le t$, а значит ответом могут быть (не теряя общности, считаем, что a идет первой):
  - (a, b, c, d) $\longrightarrow$ = $a/b + c/d = (a/c) * (c/b) + (b/d) * (c/b) = (a/c + b/d) * (c/b) \ge a/c + b/d$.
  - (a, c, b, d) $\longrightarrow$ = $a/c + b/d$.
  - (a, d, b, c) $\longrightarrow$ = $a/d + b/c \ge a/c + b/d$, так как при умножении на cd они дадут очевидно сравниваемые числа.

Решение 2

- Таким образом, чтобы получить ответ, достаточно было любым способом отсортировать эти 4 числа по возрастанию и вывести их в порядке (первое, третье, второе, четвертое).

```
print('Спасибо за внимание')
```

Материалы олимпиады

http://neerc.ifmo.ru/school