Московская командная олимпиада 2013

Разбор задач

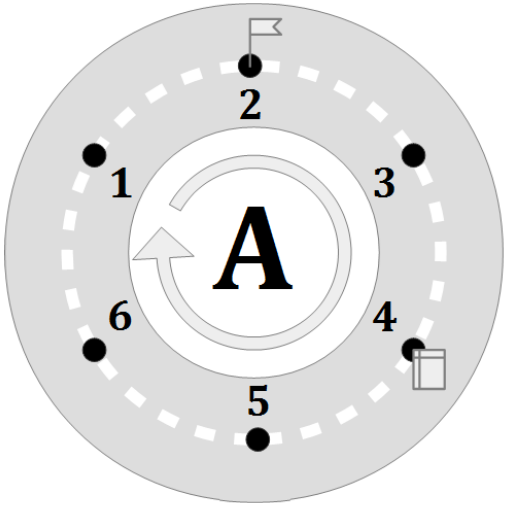
Задача «Подсчёт столбов»

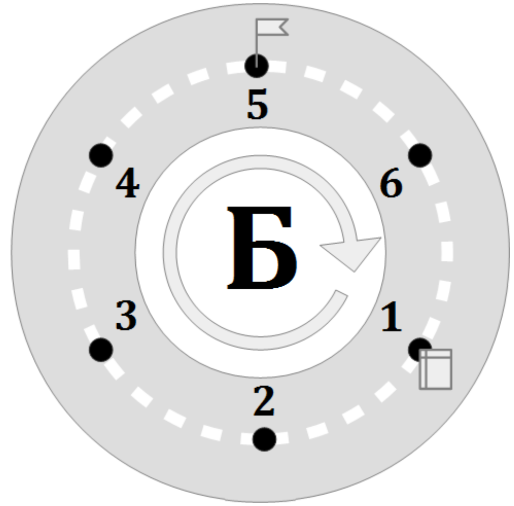
Автор идеи: Кенгуру

Разработчик и автор разбора: Антон Полднев

$Ap = 4$

$Af = 2$





$Bp = 1$

$Bf = 5$

- Если движемся в одном направлении:

$Ap - Af \equiv Bp - Bf \pmod{N} \Leftrightarrow (Ap - Af) - (Bp - Bf) \, \vdots \, N$

- N подходит, если больше $M = \max(Ap, Af, Bp, Bf)$ и является делителем $G = |(Ap - Af) - (Bp - Bf)|$.
- Перебирать все делители не нужно, так как $Gmax = 2M \Rightarrow$ либо $N = G$, либо ответа нет.
- Обратное направление: $G = |(Ap - Af) + (Bp - Bf)|$.
- Тонкости: вычесть 1, $G = 0$, переполнение.

Задача “В гору пойдет!”

Автор задачи: Роман Гусарев

Разработчик: Дмитрий Горбунов

Автор разбора: Михаил Пядеркин

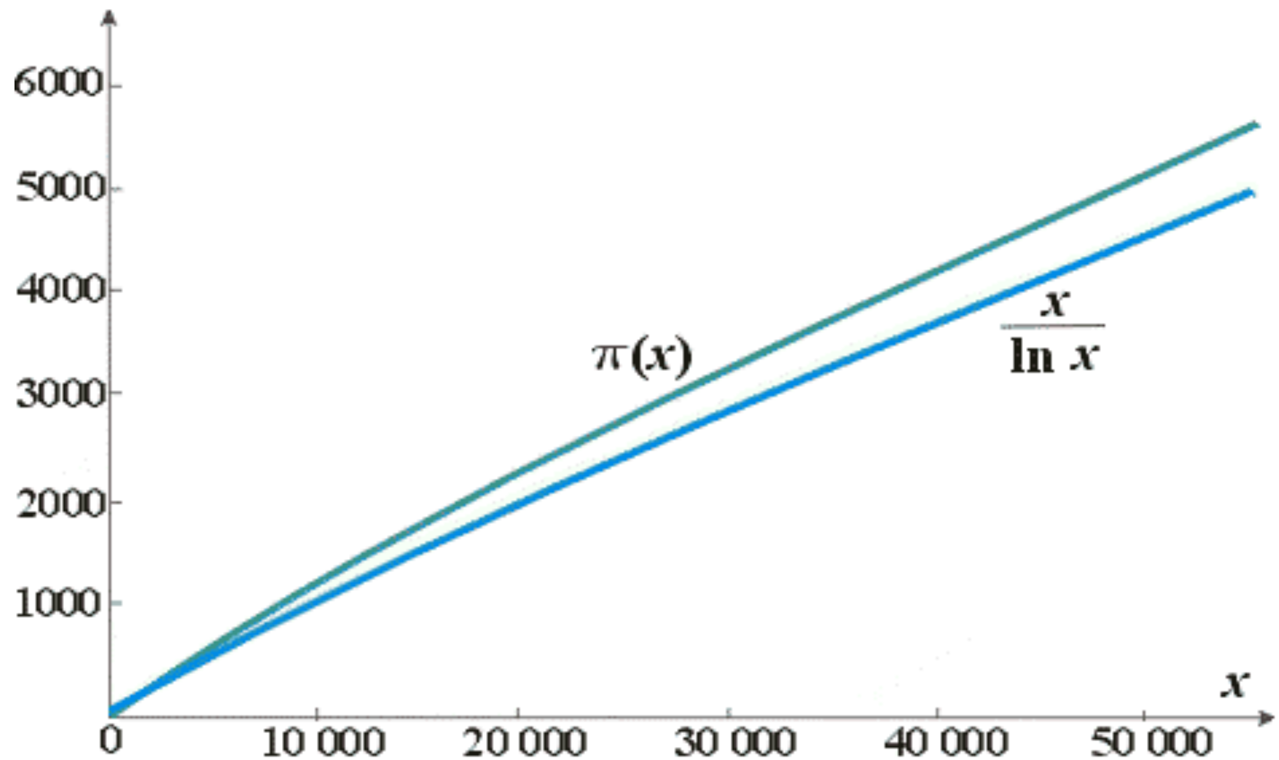
Будем идти слева направо.

Если очередной знак - “больше”, то поставим перед ним максимальное еще не поставленное число R.

Если же знак - “меньше”, то поставим перед ним минимальное еще не использованное число L.

Поддерживаем их по ходу, изначально L=1, R=n.

Задача “Проще простого”



Автор задачи и разработчик: Александр Чистяков

Автор разбора: Максим Ахмедов

- При фиксированном L количество простых на отрезке приблизительно убывает.
- При сдвиге окошка на 1 количество простых на нём меняется не более, чем на 1.
- Пусть F(x) - количество простых на отрезке [x, x + L - 1].
- Если мы найдём такие точки a и b, что F (a) >= k и F(b) <= k, то можно писать бинарный поиск (хоть функция и не монотонна!)

```
// инвариант: f(a) >= k, f(b) <= k
while b - a > 1:
| mid = (a + b) / 2
| val = f(mid)
| if (v >= k)
| | a = mid
| else
| | b = mid
```

- Осталось найти такие b и a.
- Ограничения подобраны таким образом, чтобы в качестве b можно было взять самый правый отрезок: на отрезке длины 100000, упирающемся в 10^7, 6135 простых.
- Среди всех отрезков фиксированной длины L максимальное количество простых на отрезке [2, L + 1].
- Если K превосходит количество простых на [2, L + 1], то ответа нет.
- Иначе - бинпоиск.

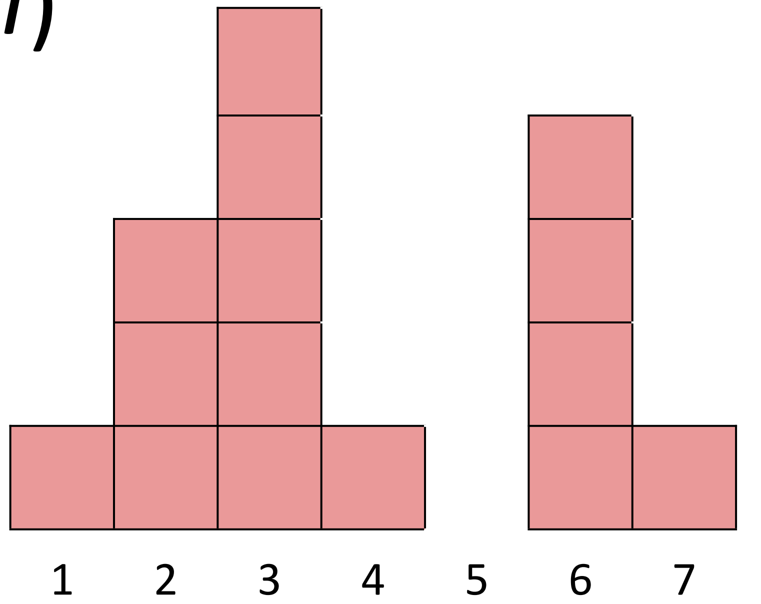
- Научимся быстро считать F(x).
- Предподсчитаем все простые до 10^7 с помощью решета Эратосфена.
- S[x] = количество простых <= x.
- Тогда F(x) = S[x + L - 1] - S[x - 1].
- Общая сложность - O(10^7 log(log(10^7)) + T * log(10^7)).

Задача «Матрёшки»

Автор задачи и разработчик: Тимур Хисматуллин

Автор разбора: Антон Полднев

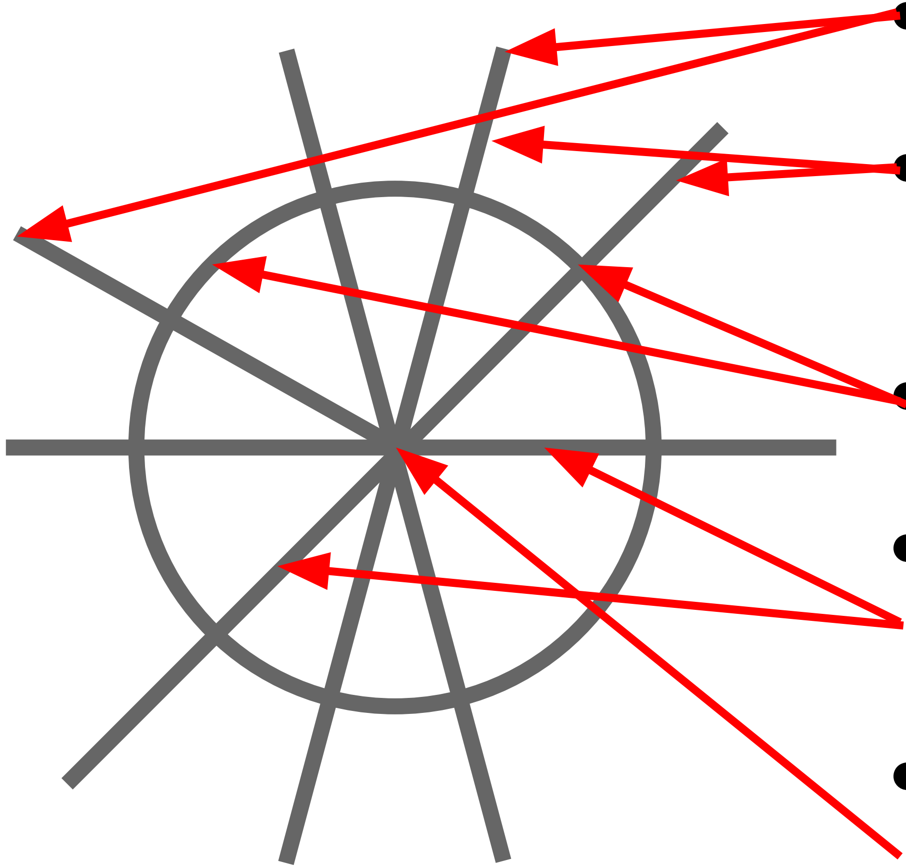
- Перейдём к количествам заготовок каждого типа — $O(N+M)$
- Выгодно жадно выделять матрёшек
- Три идеи:
  1) Стек — $O(Ans)$
  2) Список ненулевых значений — $O(N+M)$
  3) Простой проход по ненулевым значениям — $O(N+M)$



Задача «Метро»

Автор задачи и разбора: Олег Мингалёв

Как идейно устроен наш граф



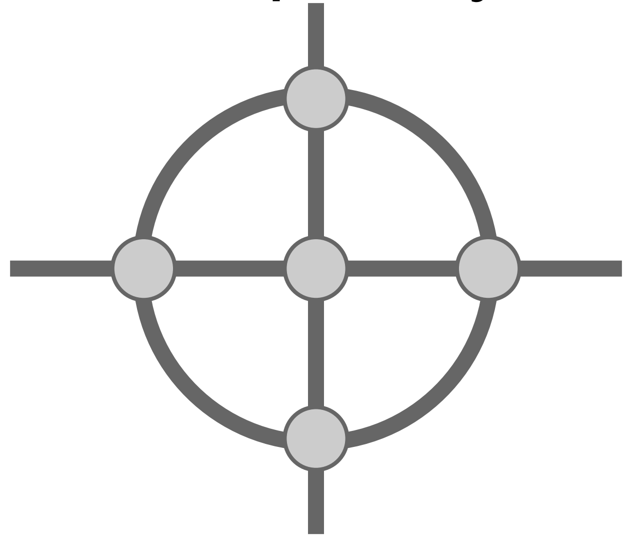
- “Листы” -- вершины степени 1
- “Внешние станции радиальных линий” -- вершини степени 2
- “Станции кольцевого метро” -- вершины степени 4
- “Внутренние станции радиальных линий” -- вершини степени 2
- “Узловая станция” -- вершина степени >= 3

Про узловую станцию

- Степень не меньше трёх
- Но все остальные вершины имеют степени 1, 2 и 4
- Таким образом, если у нас есть вершина со степенью 3 или не меньшей 5, то она должна быть узловой
- Если таких вершин несколько -- этот граф уныл до невозможности

Про узловую станцию

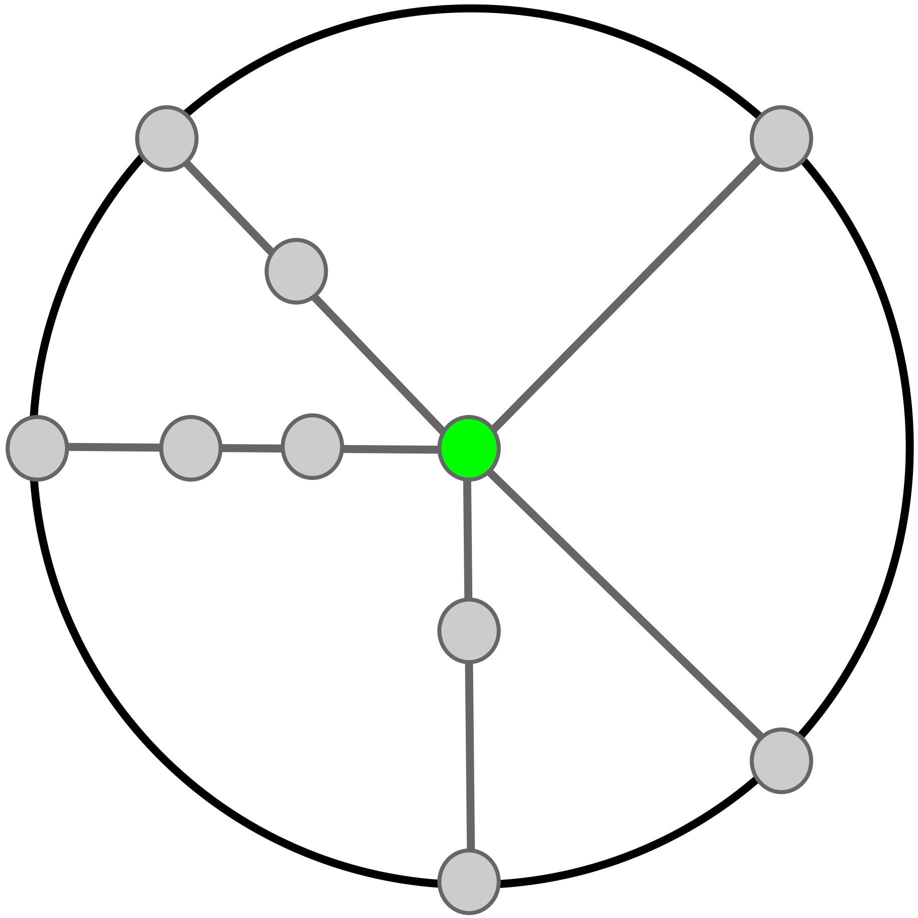
- Остался случай с узловой вершиной степени 4
- Но таких вершин в доброполезном графе должно быть в точности 5!
- Попробуем все на роль узловой



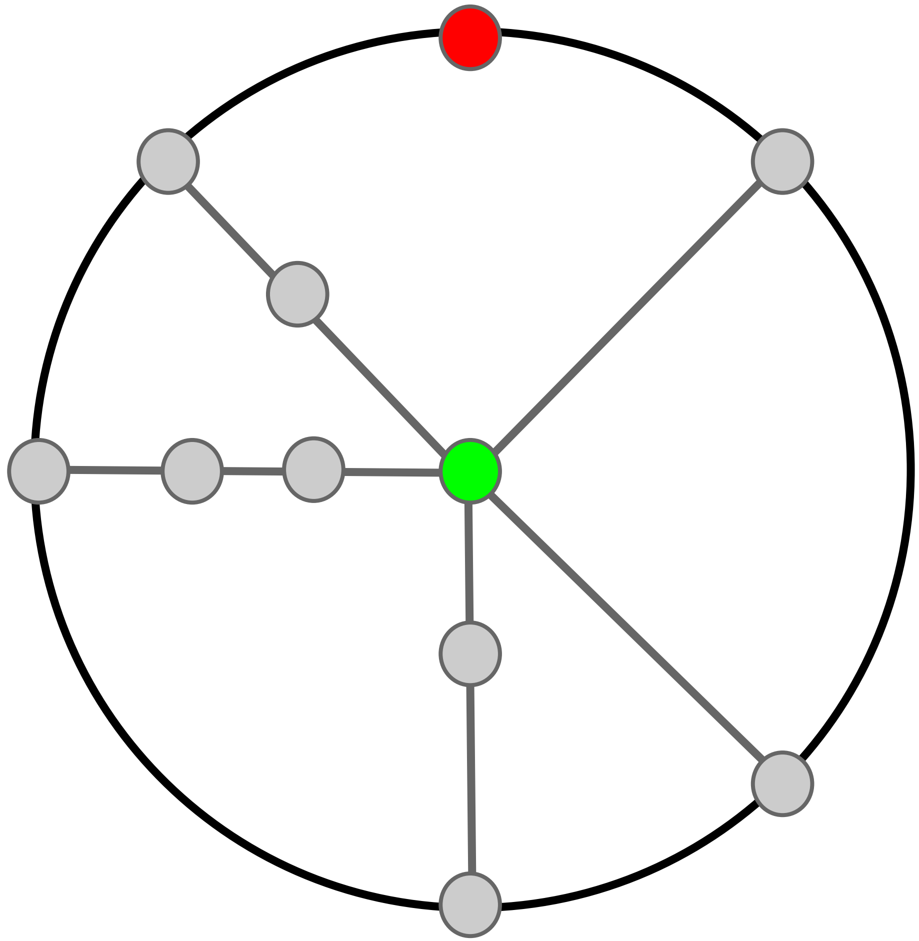
Продолжаем проверять граф

- Итак, у нас выделена узловая вершина
- Проверим граф на связность
- Будем удалять висячие линии -- листы и внешние станции радиальных лини й.
- От каждой кольцевой станции мы должны были оторвать ровно по одной висячей линии

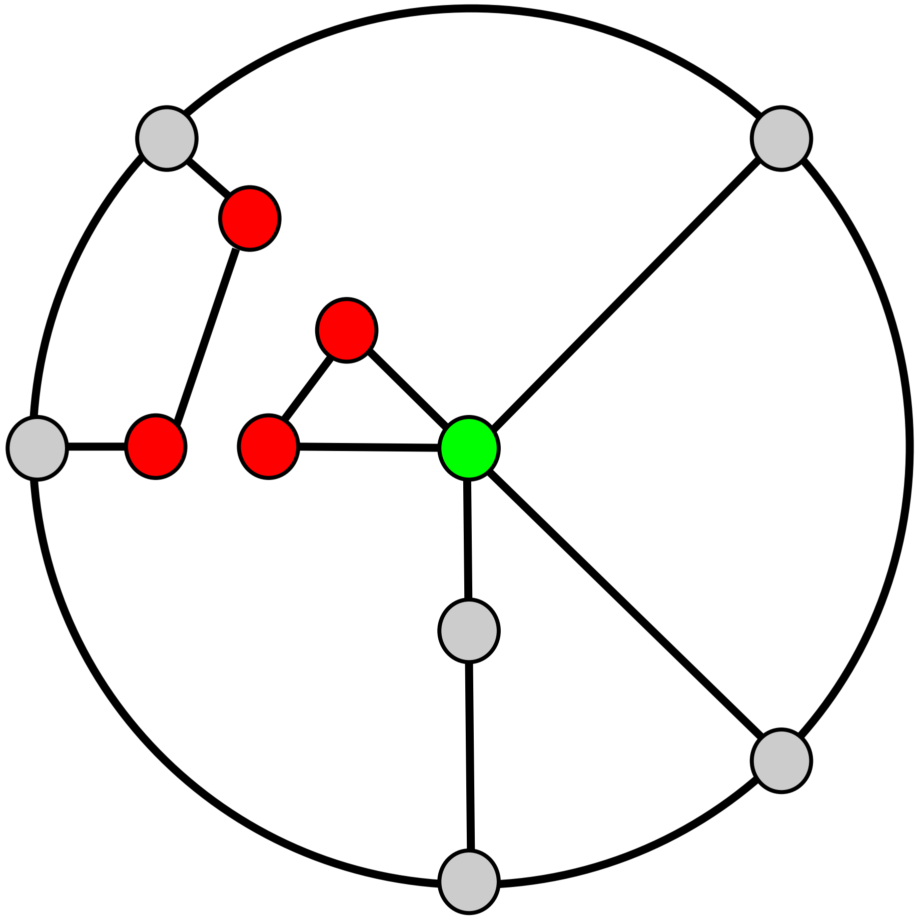
Промежуточный граф



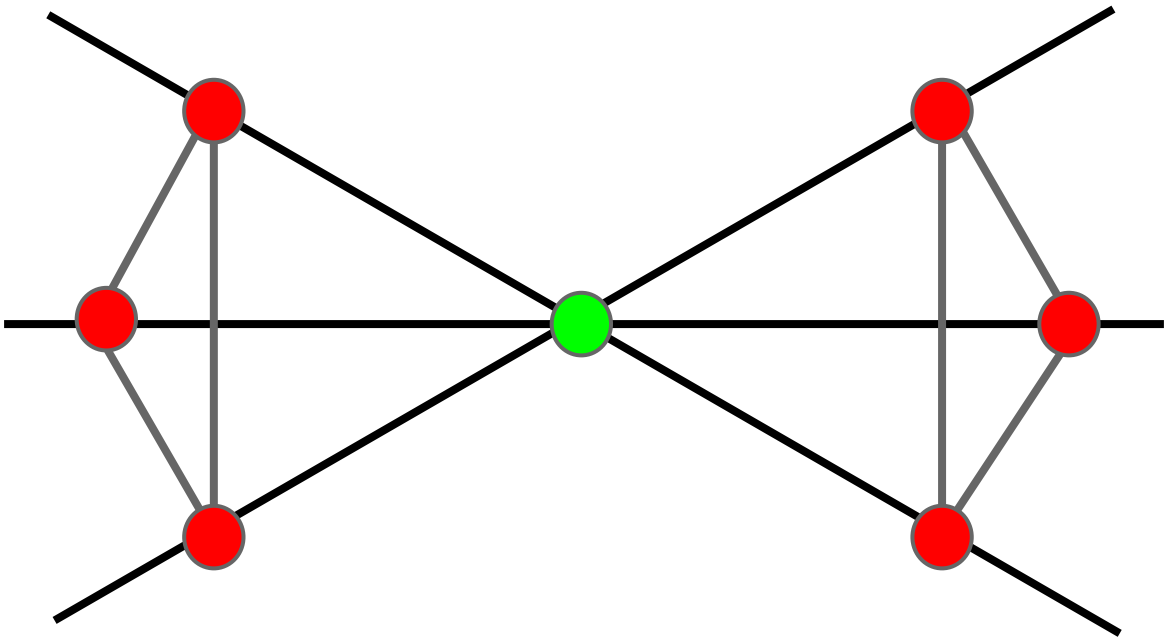
Неприятные случаи, один неприятнее другого



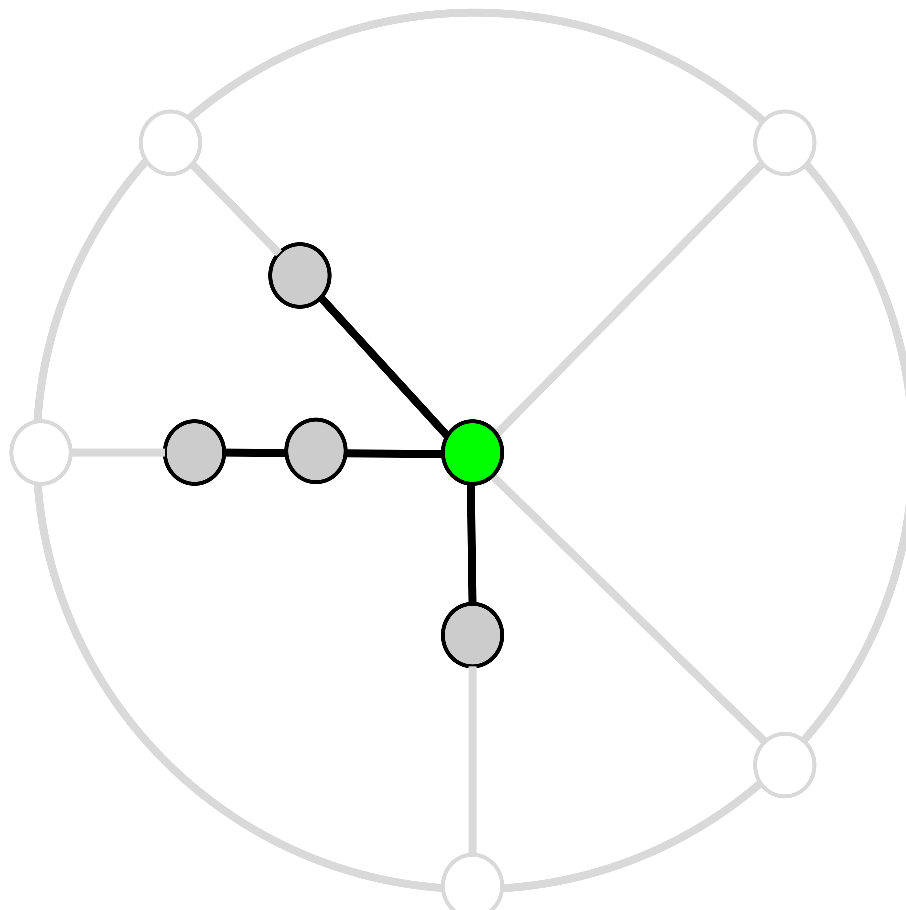
Неприятные случаи, один неприятнее другого



The best one



Граф без внешних и кольцевых станций



Узловая станция с внутренними станциями

Итого имеем

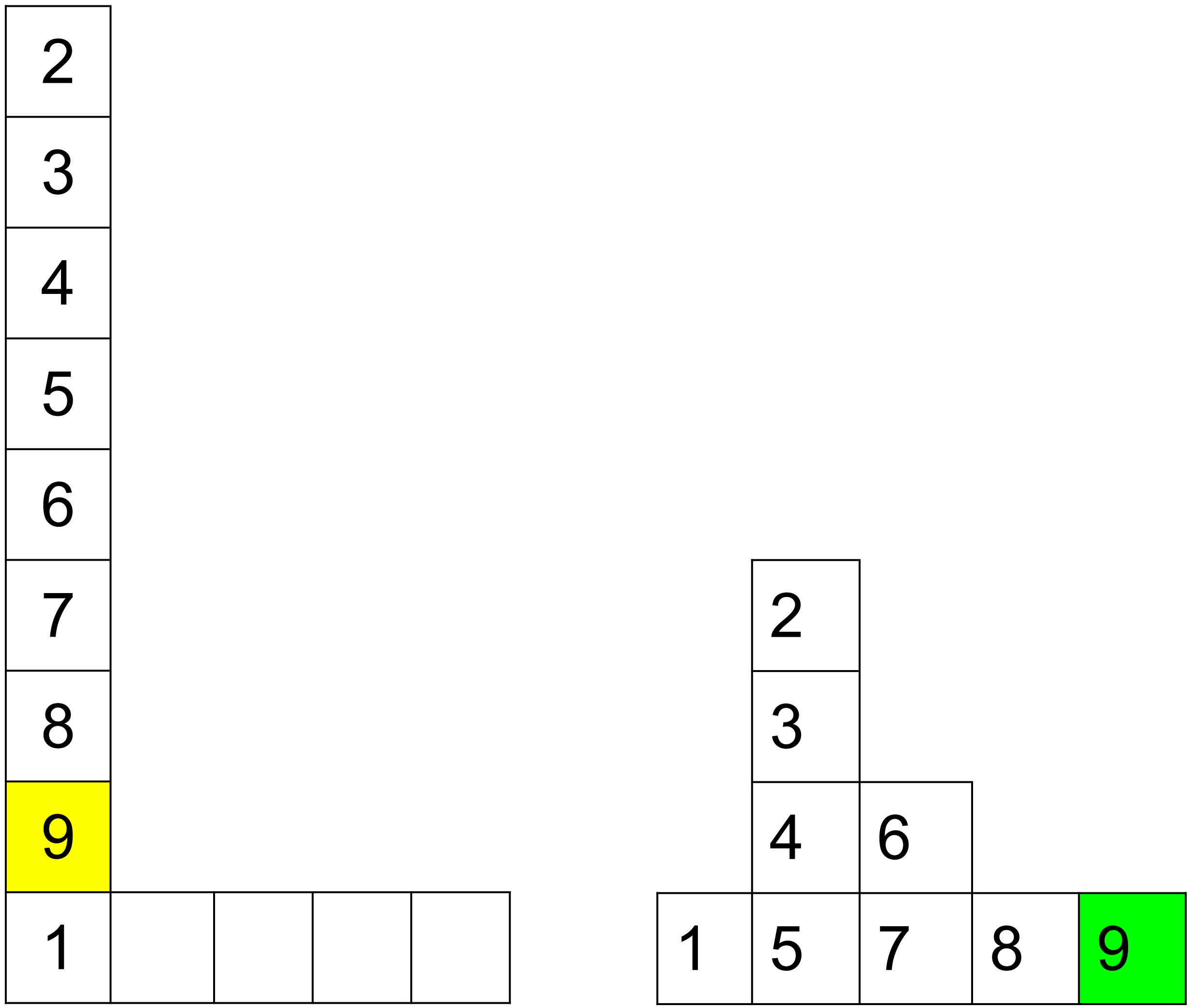
- Проверить на связность -- O(N+M)
- Откусить все внешние станции -- O(N)
- Проверить кольцевые станции на принадлежность одной станции -- O(N)
- Выбросить кольцевые вершины -- O(N)
- Откусить внутренние вершины -- O(N)

- Итоговая сложность -- O(N+M)

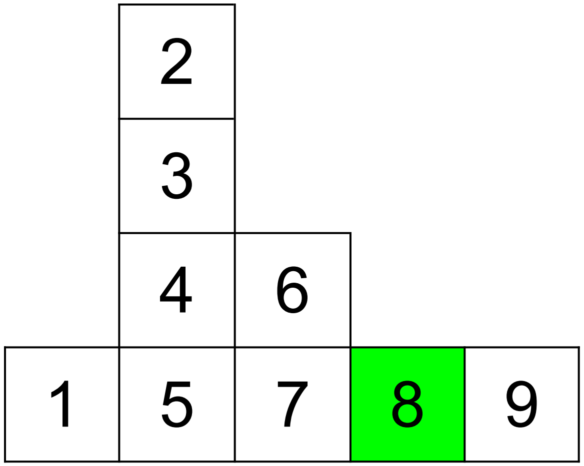
Задача “Свободная ячейка”

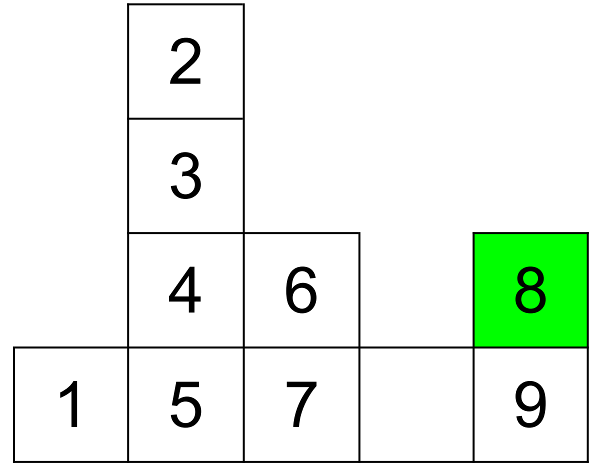
Автор задачи и разбора — Елена Андреева, разработчики — Тимур Гарипов и Елена Андреева

Решение при F = 0



Какое максимальное число карт можно убрать, чтобы освободить карту достоинством K?





Пустые слоты оставлять не имело смысла.

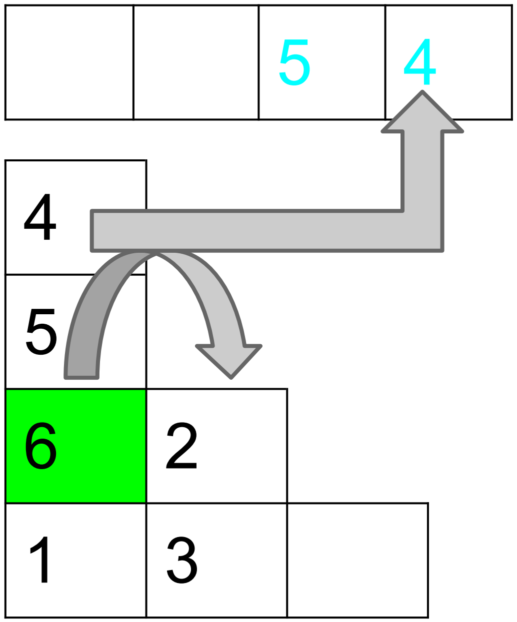
$K - 1 = 1+g[0]+g[1]+...+g[N-2]$

$g[i]$ - максимальное число карт, которое можно переложить при i свободных слотах

$g[0] = 1; g[1] = 2; g[i] = 1+g[0]+...+g[i-1]= 2\hat{}i$

$K = 1 + 1 + 1 + 2 + 4 + \ldots 2\hat{}(N-2) = 2\hat{}(N-1)+1$

F>0



Дополнительные ячейки выгодно заполнять в конце

K - 1 = F + 1 + g[0][F] + g[1][F] + … + g[N-2][F]

g[i][F] = g[i-1][F] + … + g[1][F] + g[0][F] + 1 + F

g[0][F] = F + 1

K = (F+1)2^(N-1) + 1, запомним их в A[N][F]

Наконец, сравниваем введенное K с A[N][F]

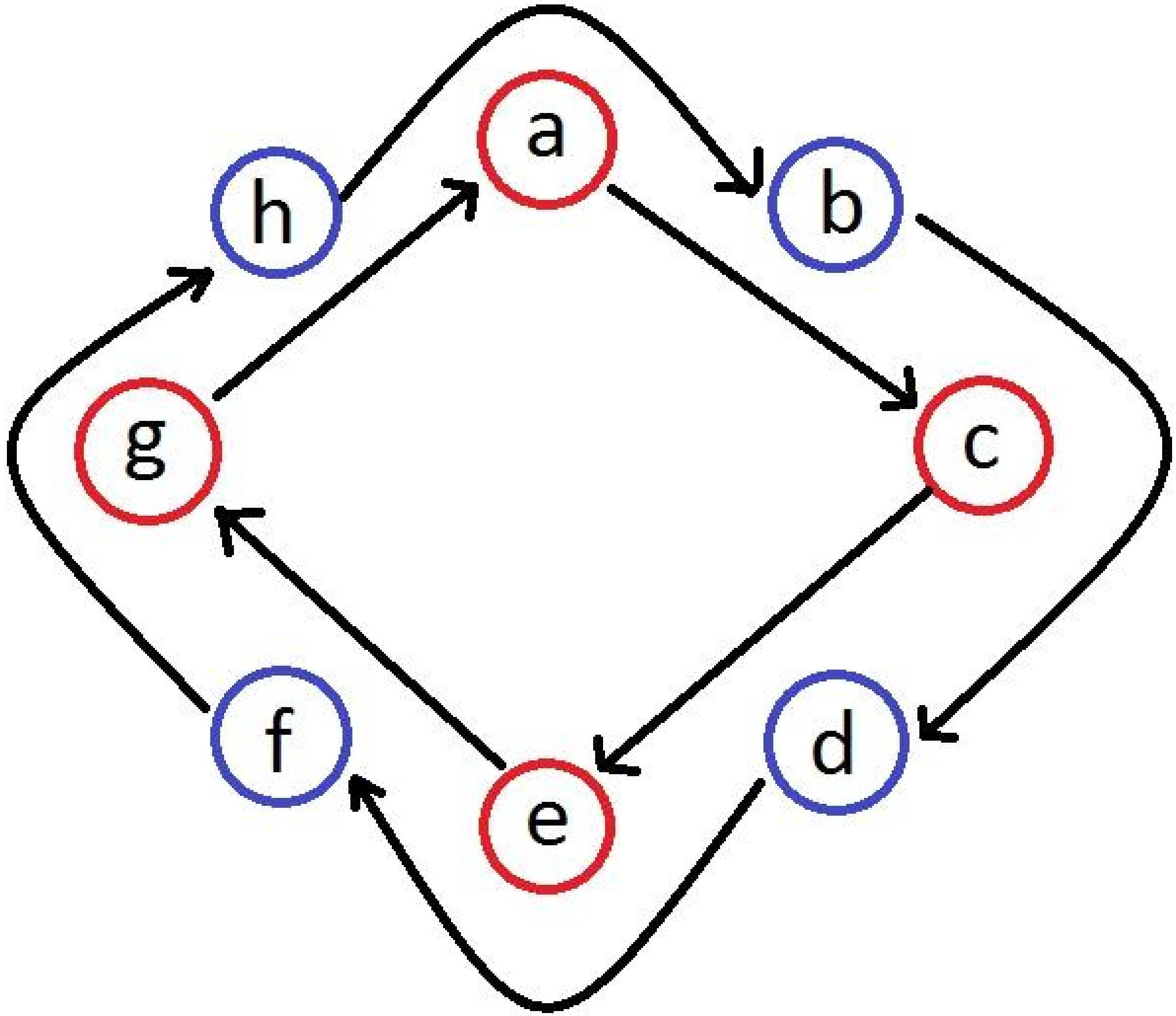
Задача “Развивающие игры”

Автор идеи - Михаил Пядеркин

Автор разбора и разработчик - Глеб Евстропов

Переформулируем задачу

1. Заметим, что строка состоит из НОД(N, |K|) независимых циклов, каждому из которых соответствует своя строка длины N / НОД(N, |K|).
2. В каждой из этих строк найдем минимальный циклический сдвиг.
3. Выберем глобальный минимум среди циклических сдвигов.
4. Выведем все сдвиги всех строк совпадающие с глобальным минимумом.



N = 8, K = 2, НОД(N, K) = 2, N / НОД(N, K) = 4.
Получаются строки: “aceg” и “bdfh”

Минимальный циклический сдвиг

1. Можно искать используя сравнение с помощью хешей. O(N log N).

2. Построить суффиксную структуру для A + A

а) Массив - первый элемент не короче |A|.

б) Дерево - самый левый (при лекс. обходе) элемент на глубине |A|.

в) Автомат - |A| шагов по мин. переходу (упражнение: доказать что столько шагов всегда можно сделать для A + A)

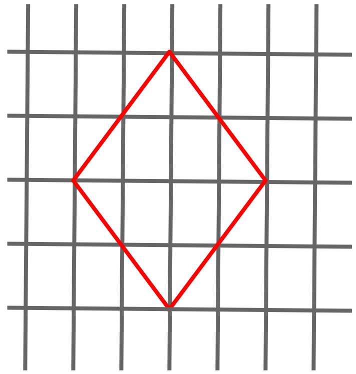
3. Жадный алгоритм Дюваля - самый короткий и простой способ.

Задача “Вечеринка в Нью-Йорке”

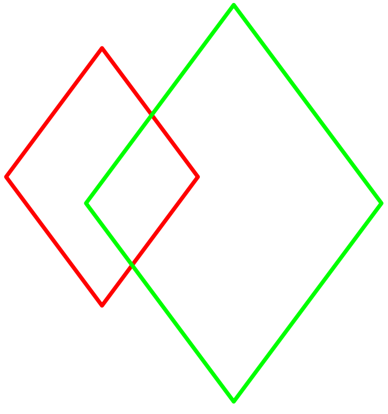
Автор задачи: Аким Кумок

Автор разбора и разработчик: Михаил Пядеркин

Заметим, что множество точек, расположенных от заданной на расстоянии не более чем L, представляет собой ромб



Можно воспользоваться бинарным поиском по ответу. После этого нужно пересечь все ромбы (это ромб), а потом найти любой хостел внутри пересечения.



Пересечение ромбов осуществляется также, как и прямоугольников.

Простое решение

“Повернем” картинку на 45 градусов.

А именно, сделаем замену координат: точке (x,y) будет соответствовать точка (x+y,x-y)

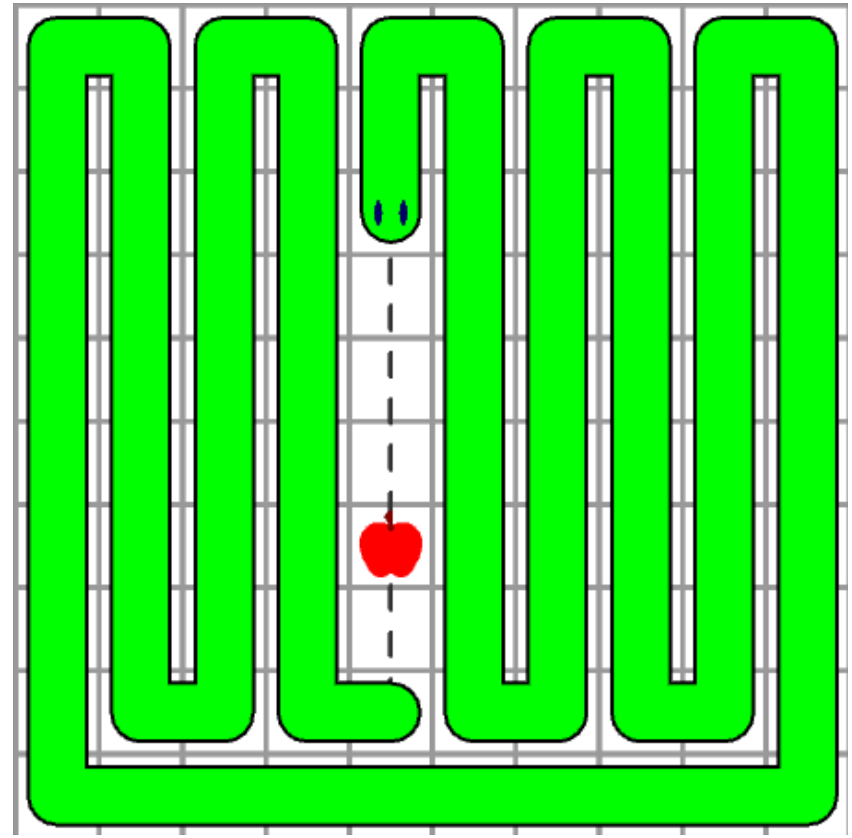
Если раньше расстояние между точками вычислялось как |x1-x2|+|y1-y2|, то в новых координатах оно вычисляется как max(|x1-x2|,|y1-y2|).

Таким образом, можно оставить лишь 4 хостела с экстремальными новыми координатами, а все отели перебрать

Задача “Змейка”

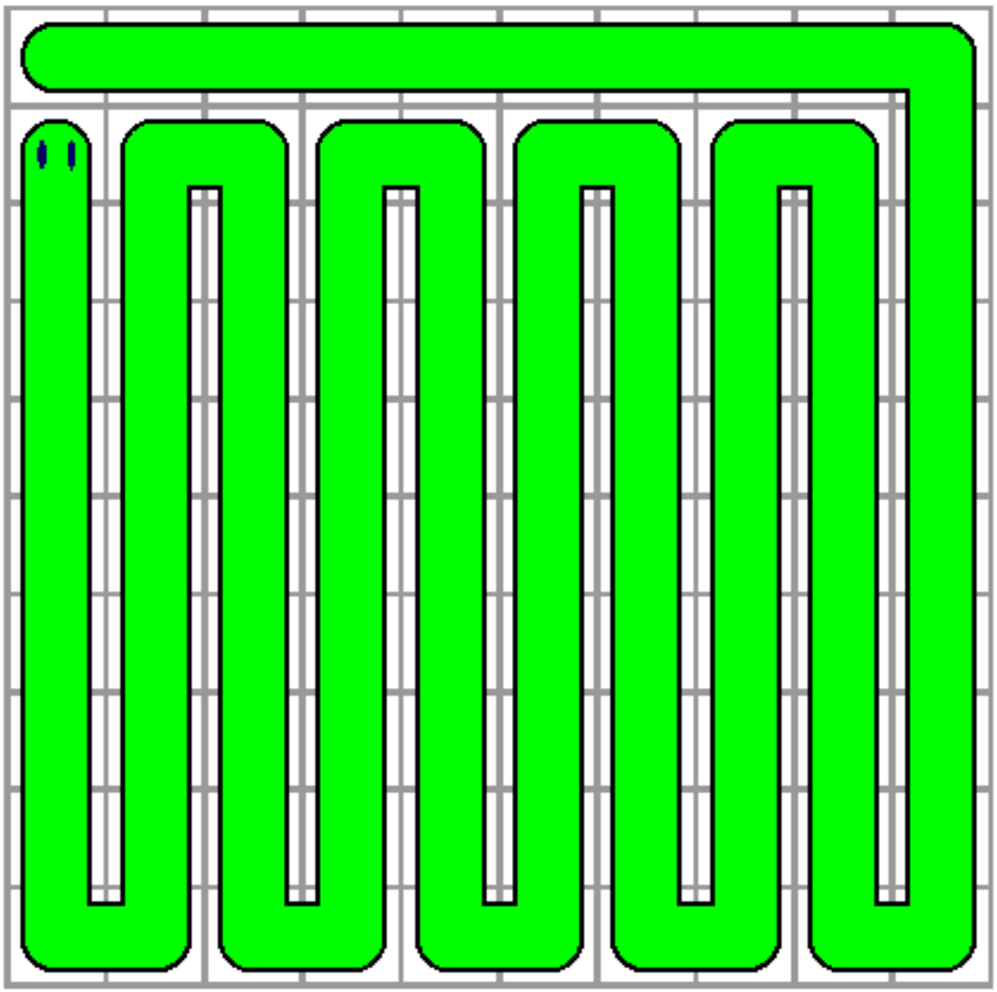
Автор задачи и разбора: Максим Ахмедов

- Жадные решения работают плохо.
- А работают плохо они при больших размерах змейки.
- При длине змейки почти в NM ей придётся ползти по своему контуру.

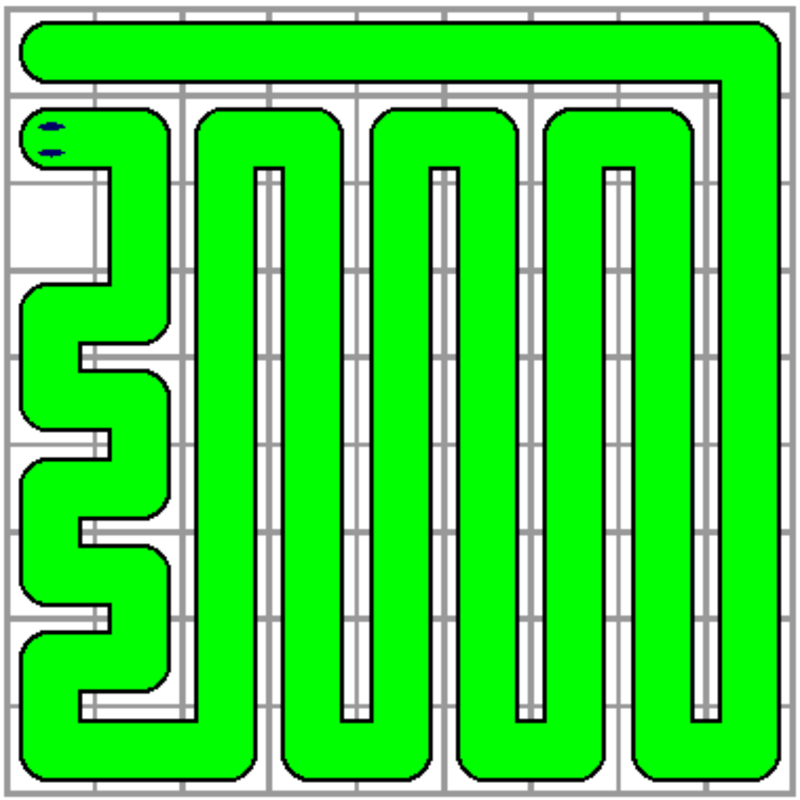


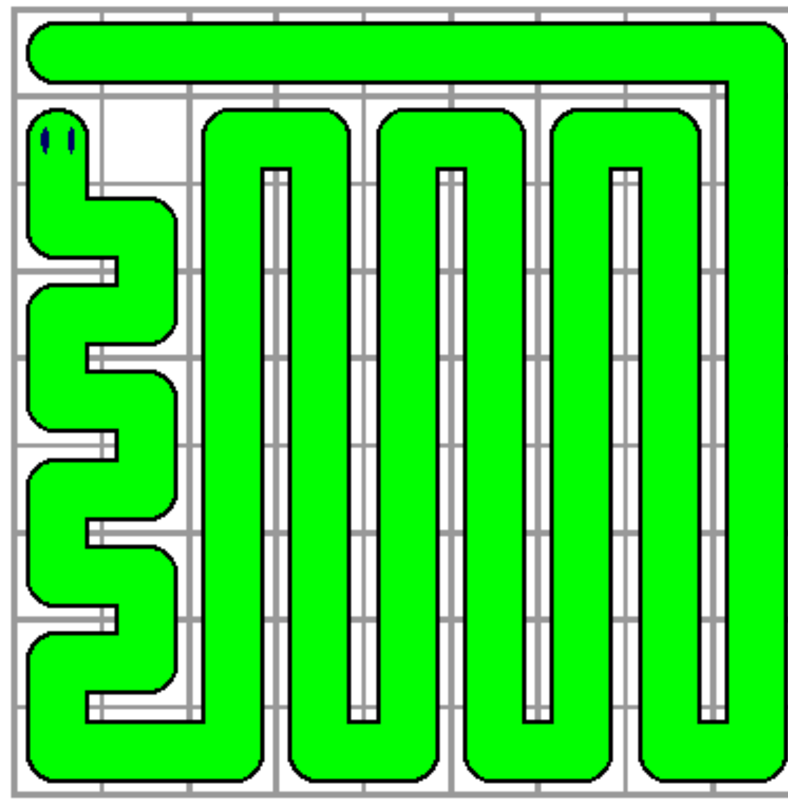
- Будем гонять змейку по одному и тому же циклическому маршруту, не обращая внимания на яблоки.

- При чётном N x M существует маршрут, посещающий каждую клетку поля ровно один раз - гамильтонов цикл.



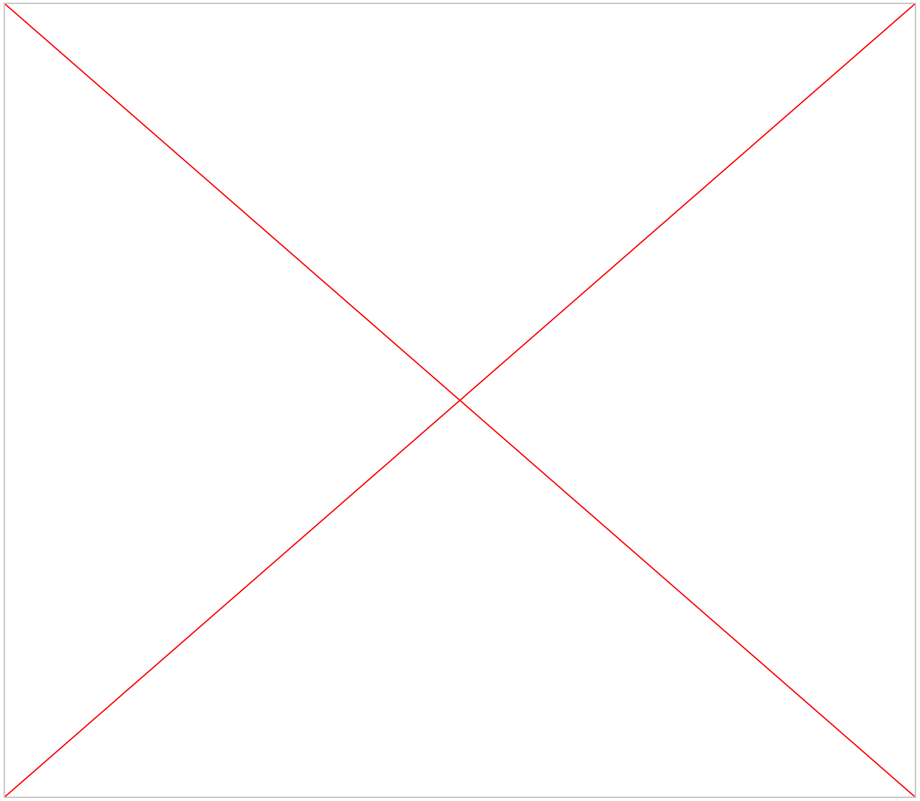
- При нечётном N x M сущетсвуют два “почти гамильтоновых цикла”, отличающихся по одной клетке.





- Максимум - N x M x Len ходов,
- Len < 2 x N x M
- Итого - 2 x 25^4 ~ 8 * 10^5 ходов. Реально - гораздо меньше.

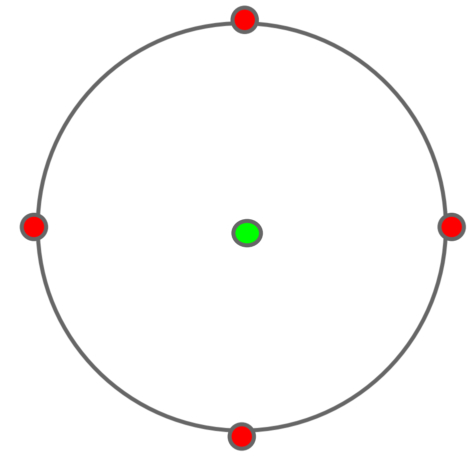
Задача “Сладкоежка”



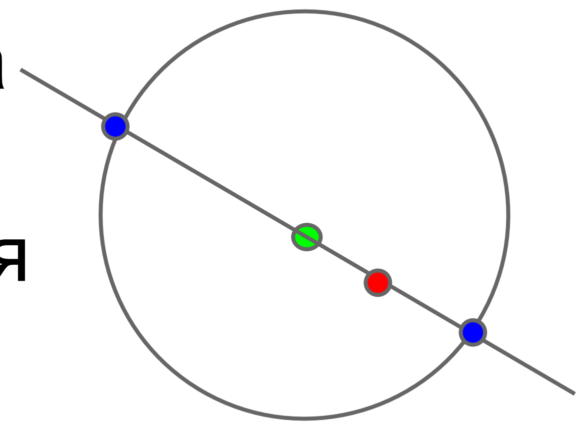
Автор задачи и разработчик: Тимофей Ющенко

Автор разбора: Михаил Пядеркин

1. Добавим фиктивные точки, чтобы не разбирать случаи, когда точек мало или они лежат на одной прямой



2. “Спроецируем” каждую точку на окружность: добавим на окружность две точки пересечения с прямой, проведенной через центр и исходную точку



3. Отсортируем полученные синие точки по углу относительно центра, удалим одинаковые (близкие)