問３
たのしいたのしい
たのしい家庭菜園

解説：村井

問題概要

（イラスト：いらすとや）

赤、緑、黄の色のいずれかの葉をつけるジョイ草がある

問題概要

同じ色の葉のジョイ草が隣り合うとよくないので、
そのようなことが起きないようにしたい

問題概要

１回の操作で、隣り合うジョイ草を入れ替えることができる

問題概要

同じ色のジョイ草が隣り合わないようにしたい

小課題 1 (5 点)

- $N \leq 15$: かなり小さい
- 赤が r 個、緑が g 個、黄が y 個の場合、並べ方は $\frac{N!}{r!g!y!}$ 通り
- $r = g = y = 5$ のときが最大で、756756 通り
- 幅優先探索 (BFS) で全探索可能

小課題 3 (15 点)

- S の各文字は R, G のいずれかである

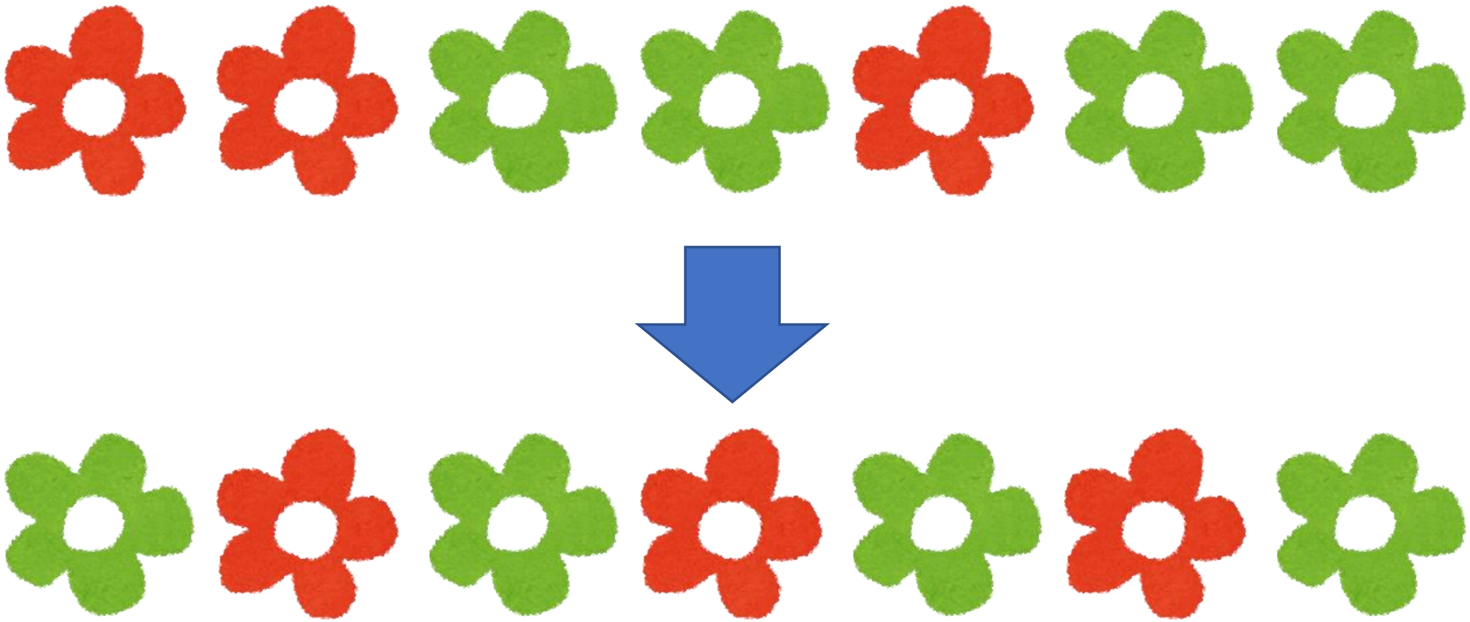
これを、並び替えて同じ色が隣り合わないようにしたい

並び替えた後の様子はかなり限られる：

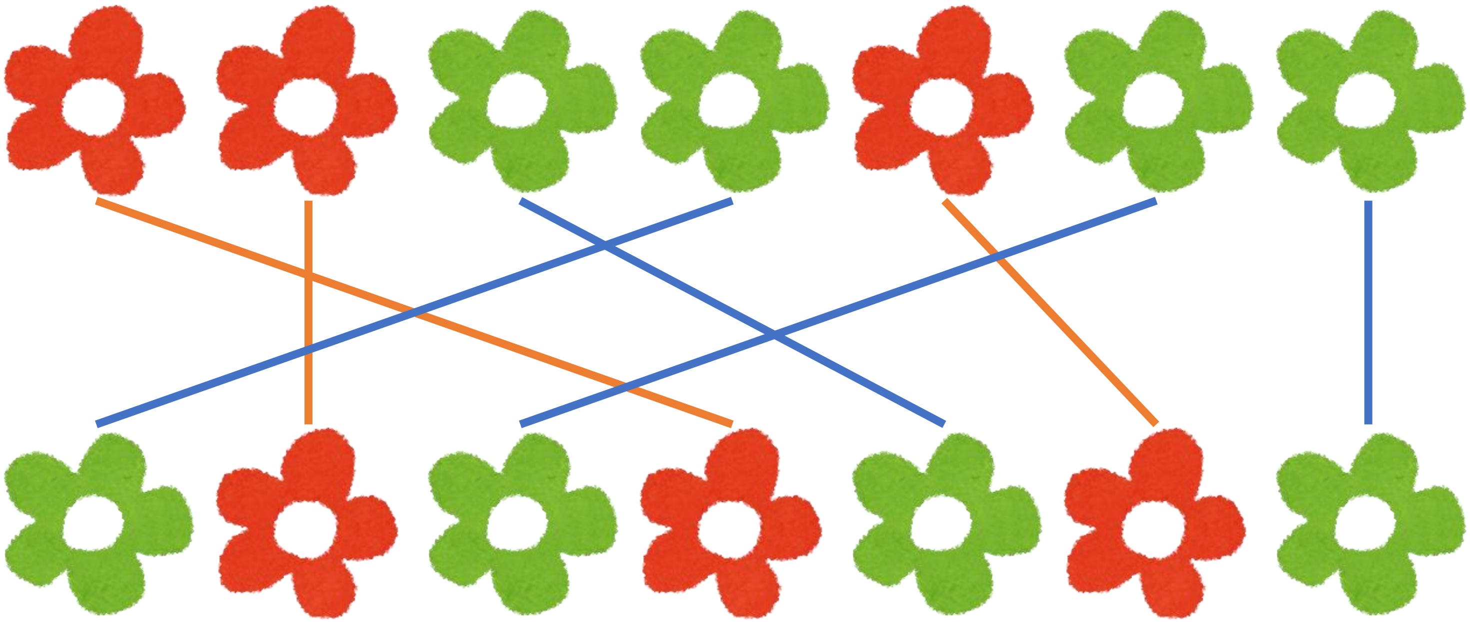
赤が最初

緑が最初

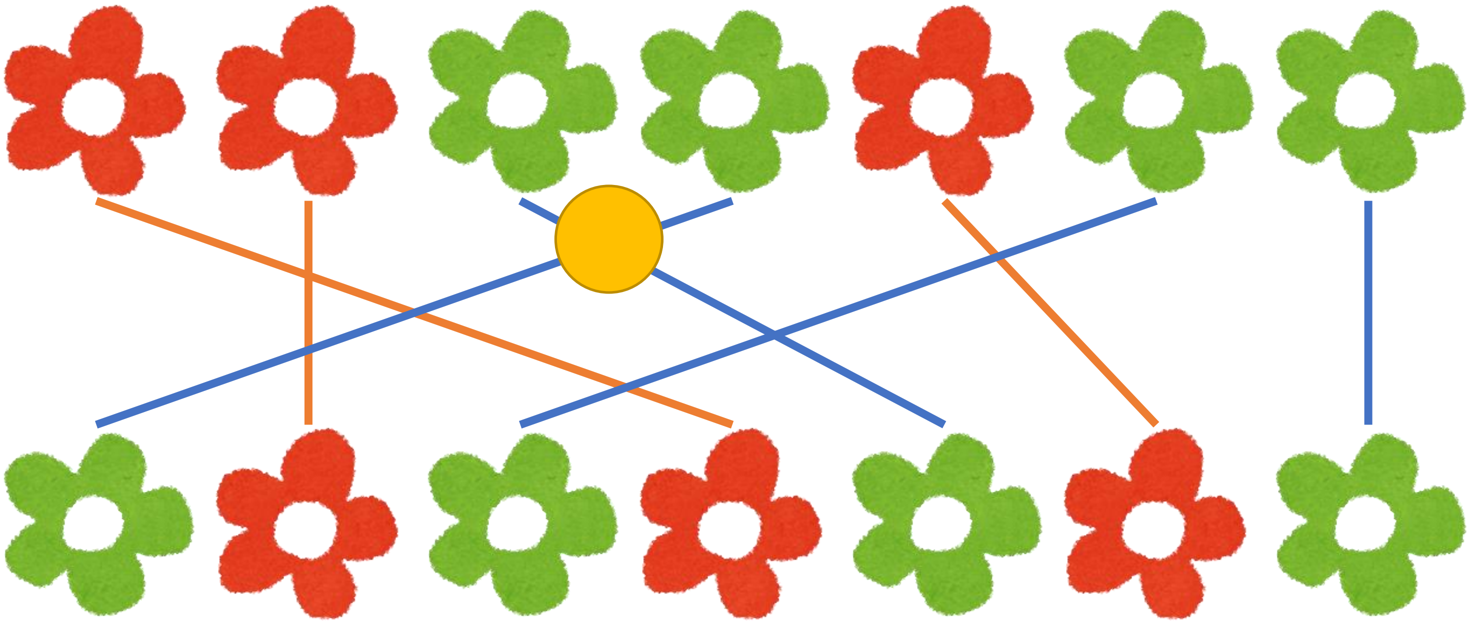
可能な目標状態が２通りだけなので、両方試せばよい？



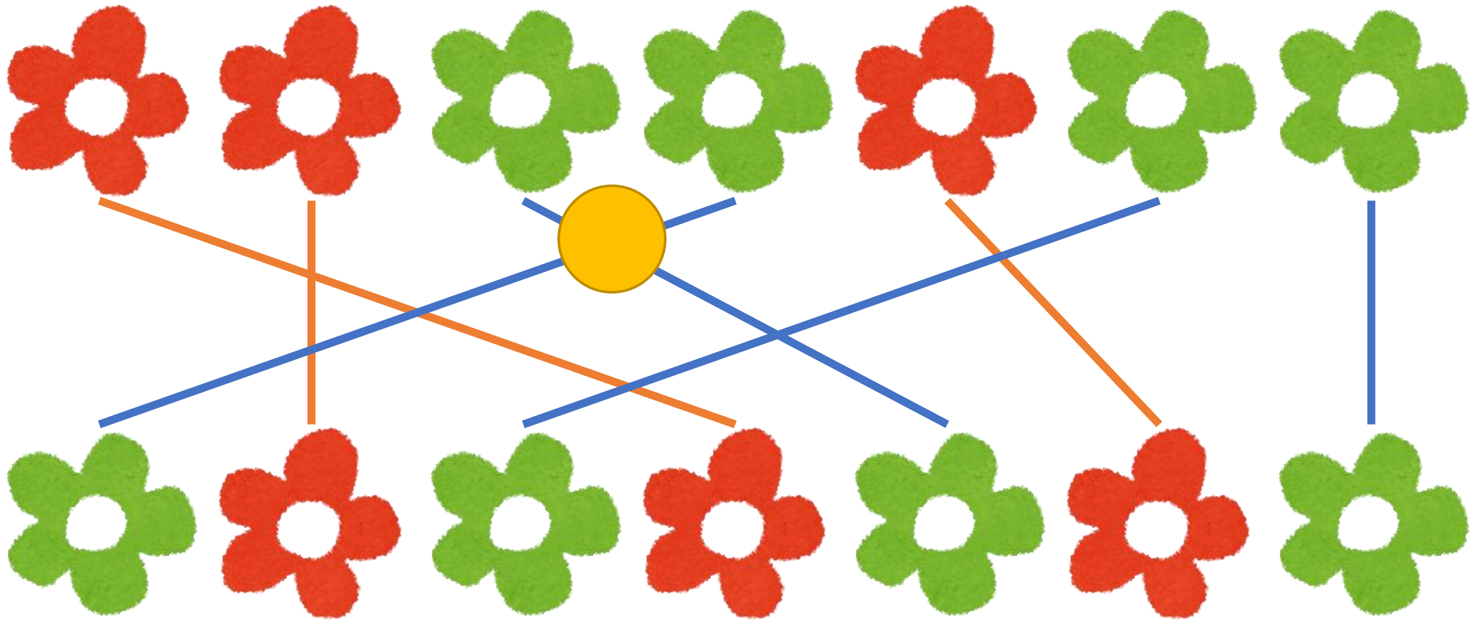
目標状態を決めたとき、
最小の操作回数は？



とりあえず、上と下でどのジョイ草が対応してるか考える

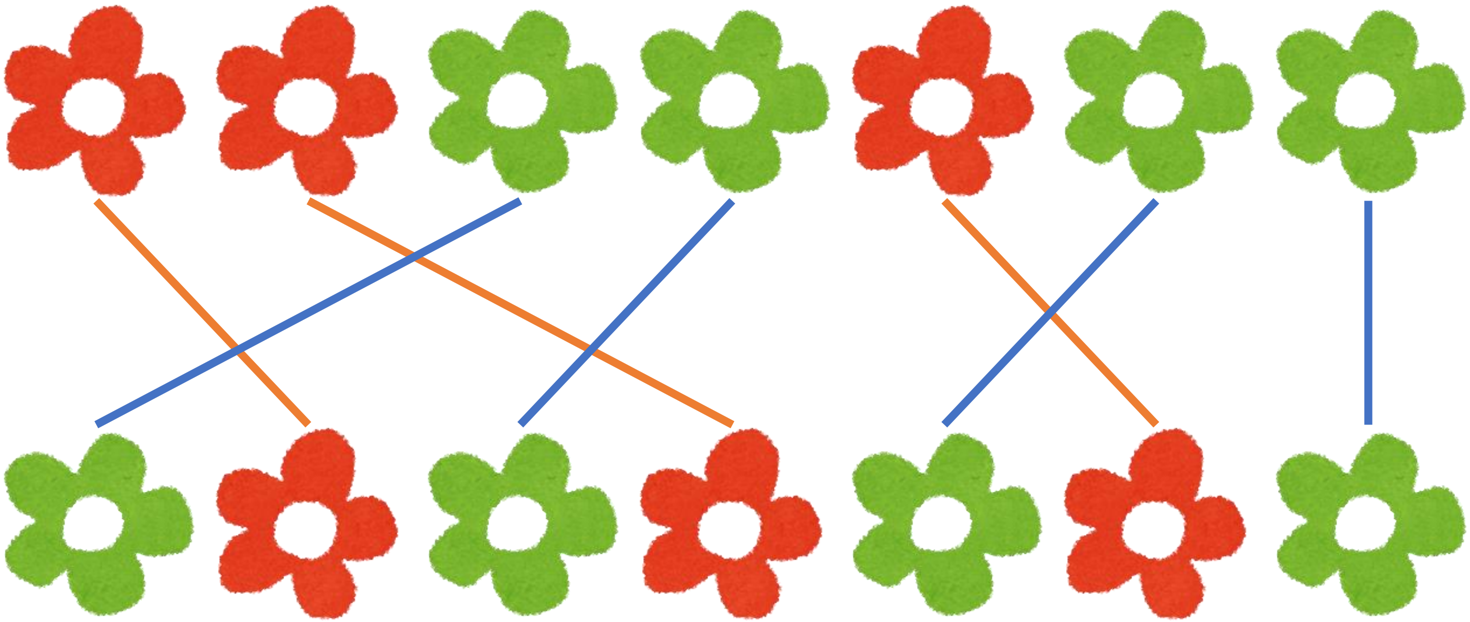


対応関係が交差している場合、その 2 つのジョイ草同士を入れ替える必要がある

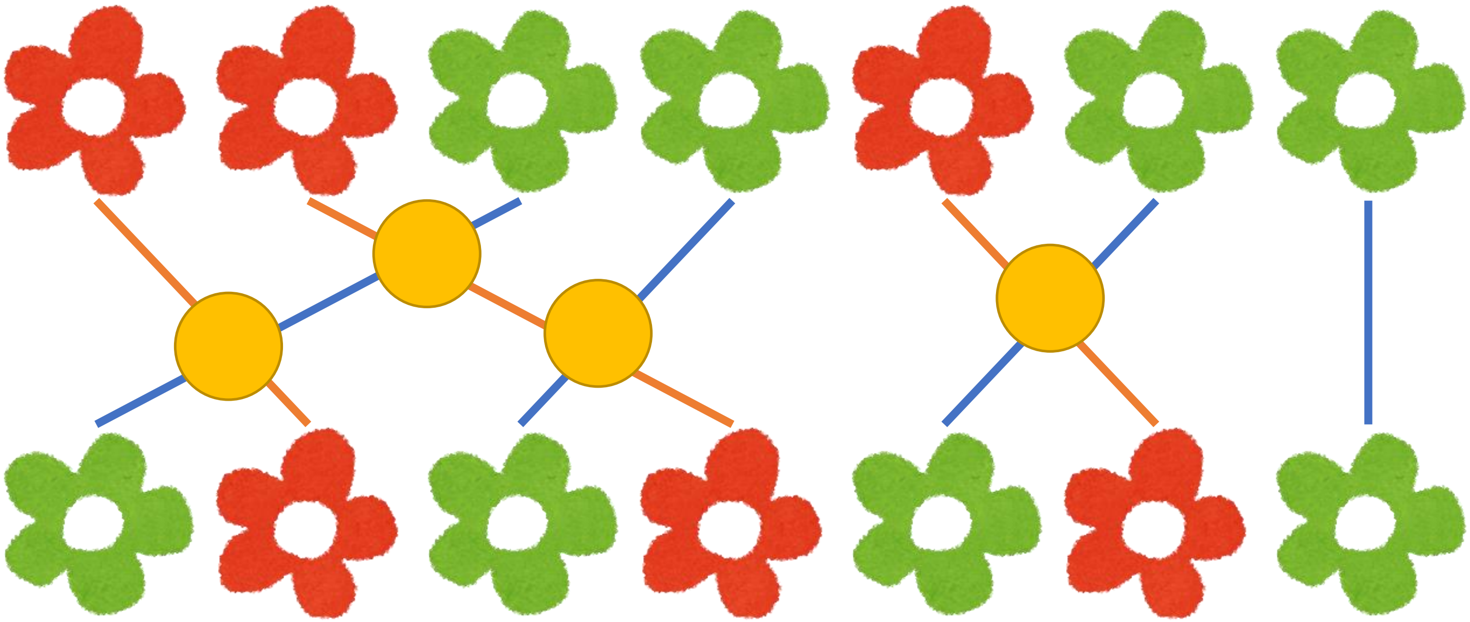


対応関係が交差している場合、その 2 つのジョイ草同士を入れ替える必要がある

でも、同じ色のジョイ草を入れ替える意味はまったくない！→最適ではない 😥

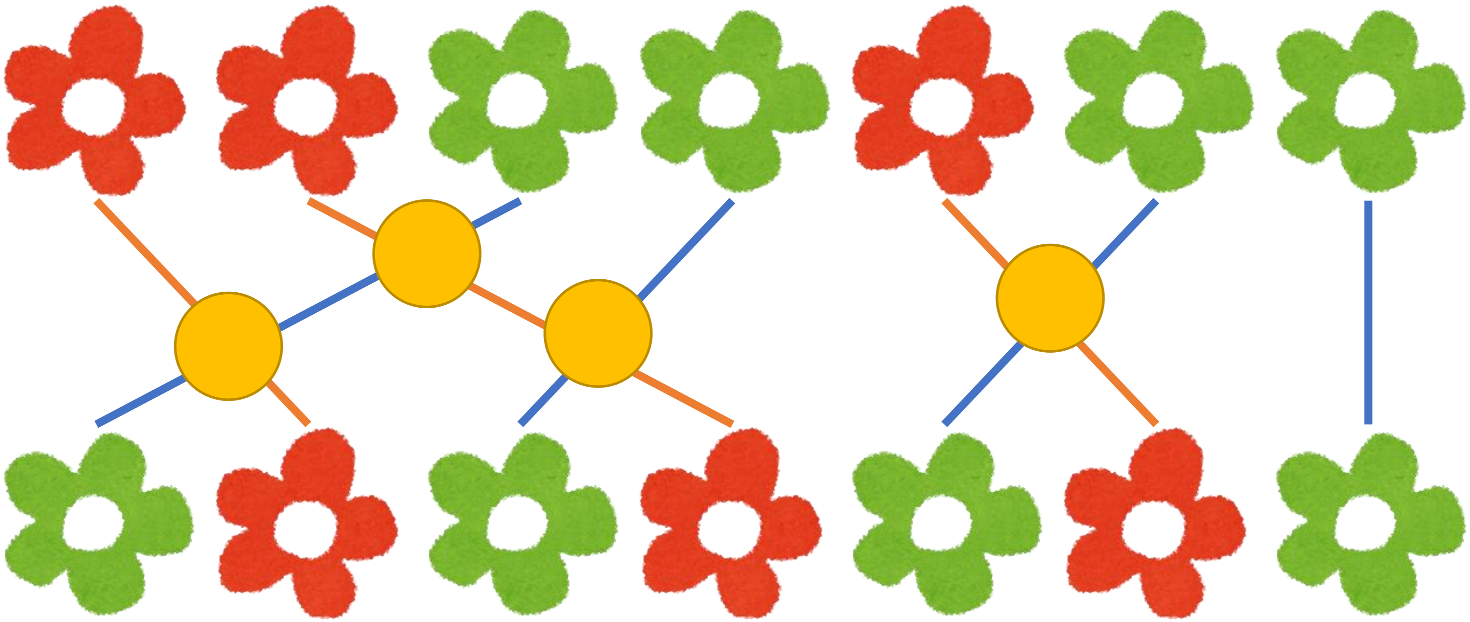


結局、同じ色の中では左から順番に対応付ける
やり方だけ考えればOK！



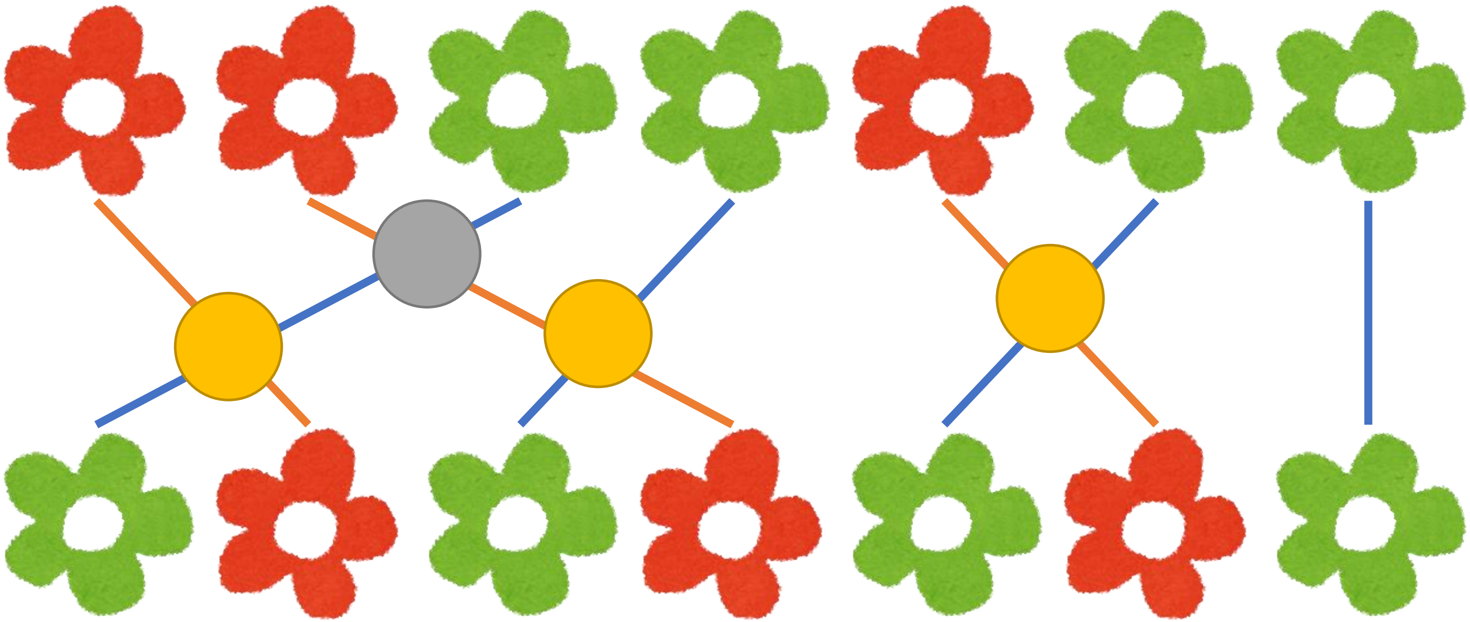
やっぱりこの交差回数くらいは最低限操作が必要
この回数で足りる？
→実は足りる！

なぜ？



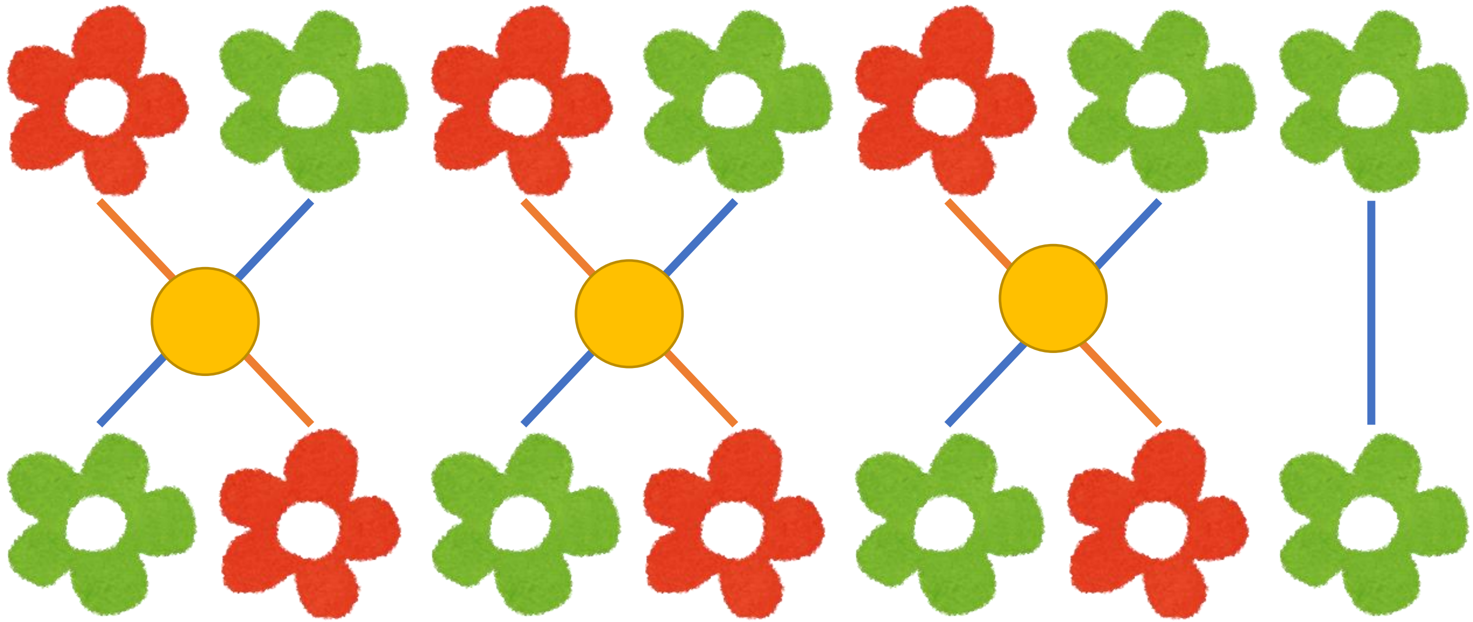
１回の操作で、交差回数を１回減らすことができる

なぜ？



1 回の操作で、交差回数を 1 回減らすことができる

なぜ？



1 回の操作で、交差回数を 1 回減らすことができる
（できないとしたら、最初から正しい順に並んでいる）

小課題 3 の解法

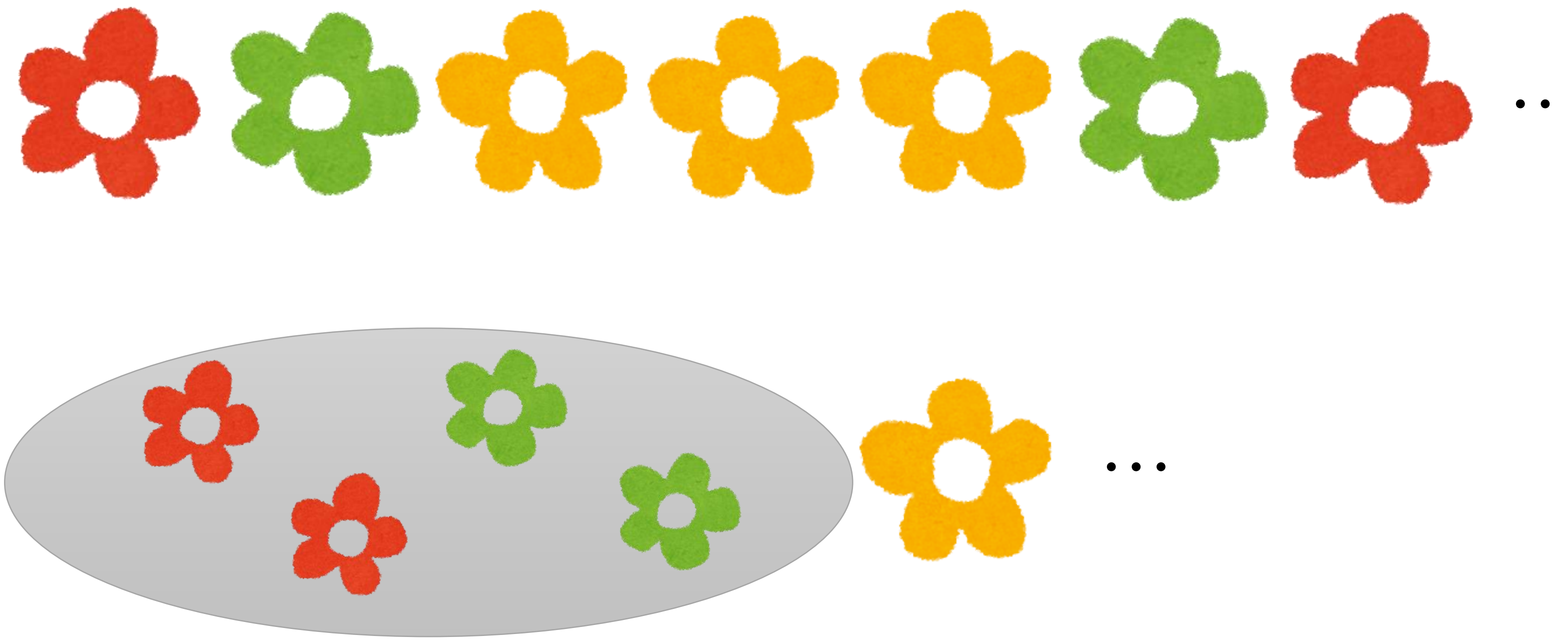
- 目標状態 2 パターンを両方試す
- そもそも赤と緑の個数が合わない場合は無理
- 個数が合っていれば、同じ色は左から順に対応させるとして、交差回数を数える

注意

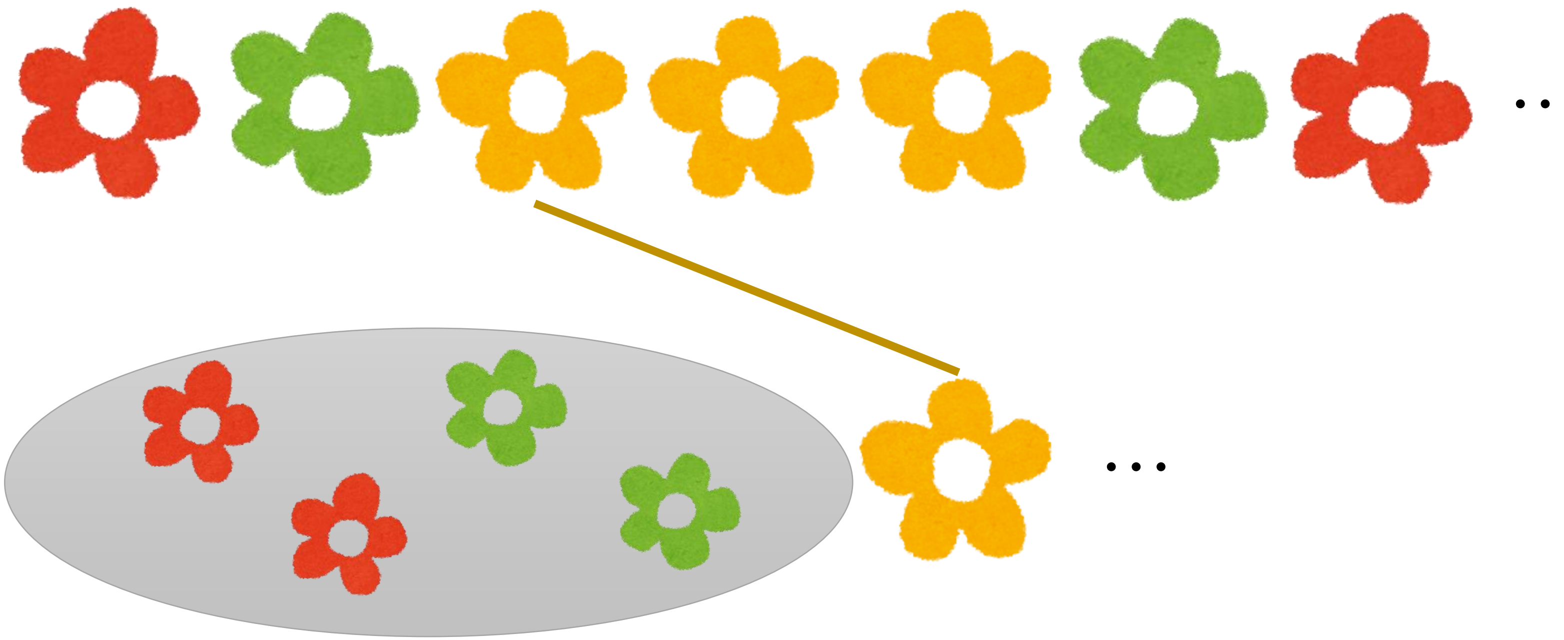
目標状態を決めてからの議論は、赤緑黄の場合にも適用可能

小課題 2 (55 点)

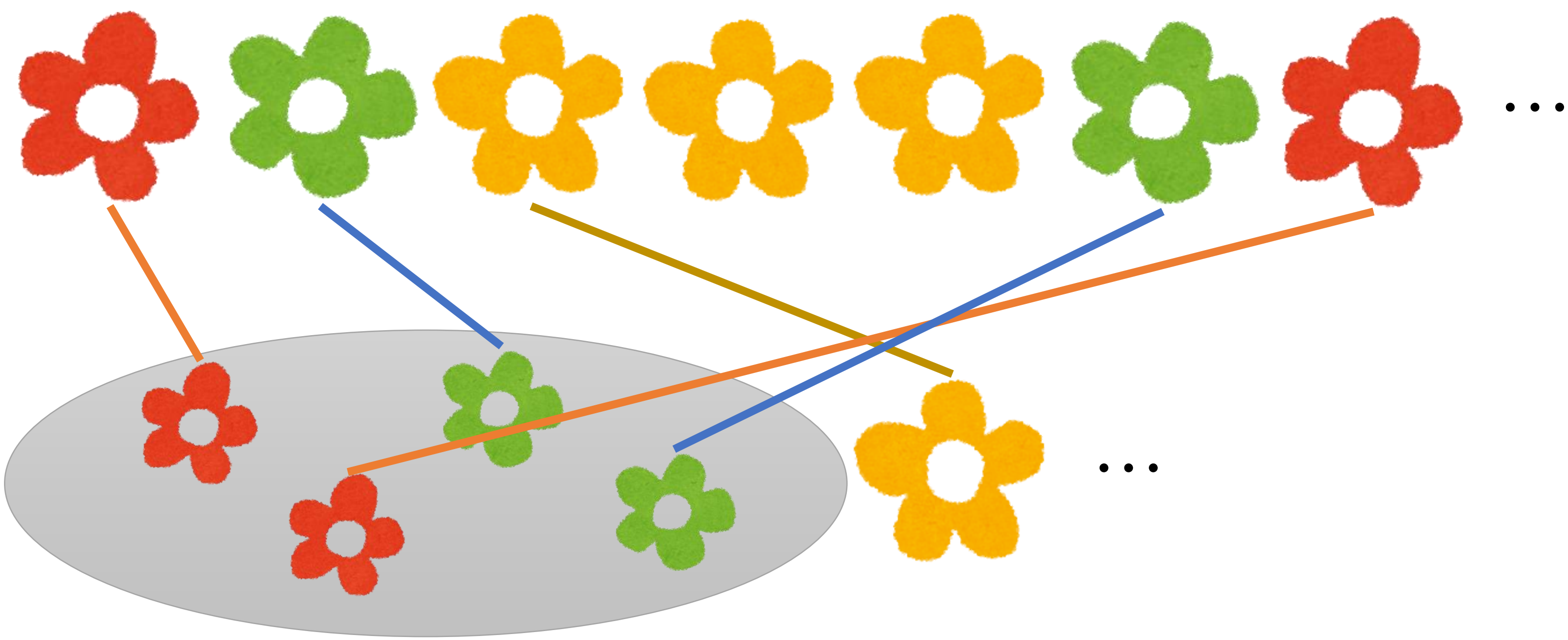
- $N \leq 60$
- 目標状態 1 個ずつ試すのはとても無理



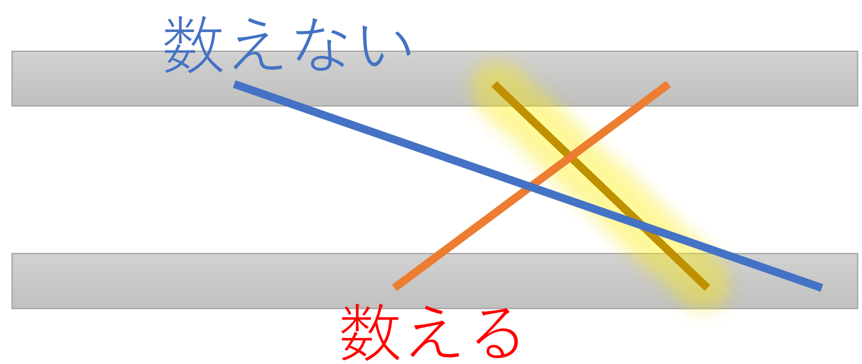
目標状態における配置を、左から順に 1 個ずつ決定していくことを考える

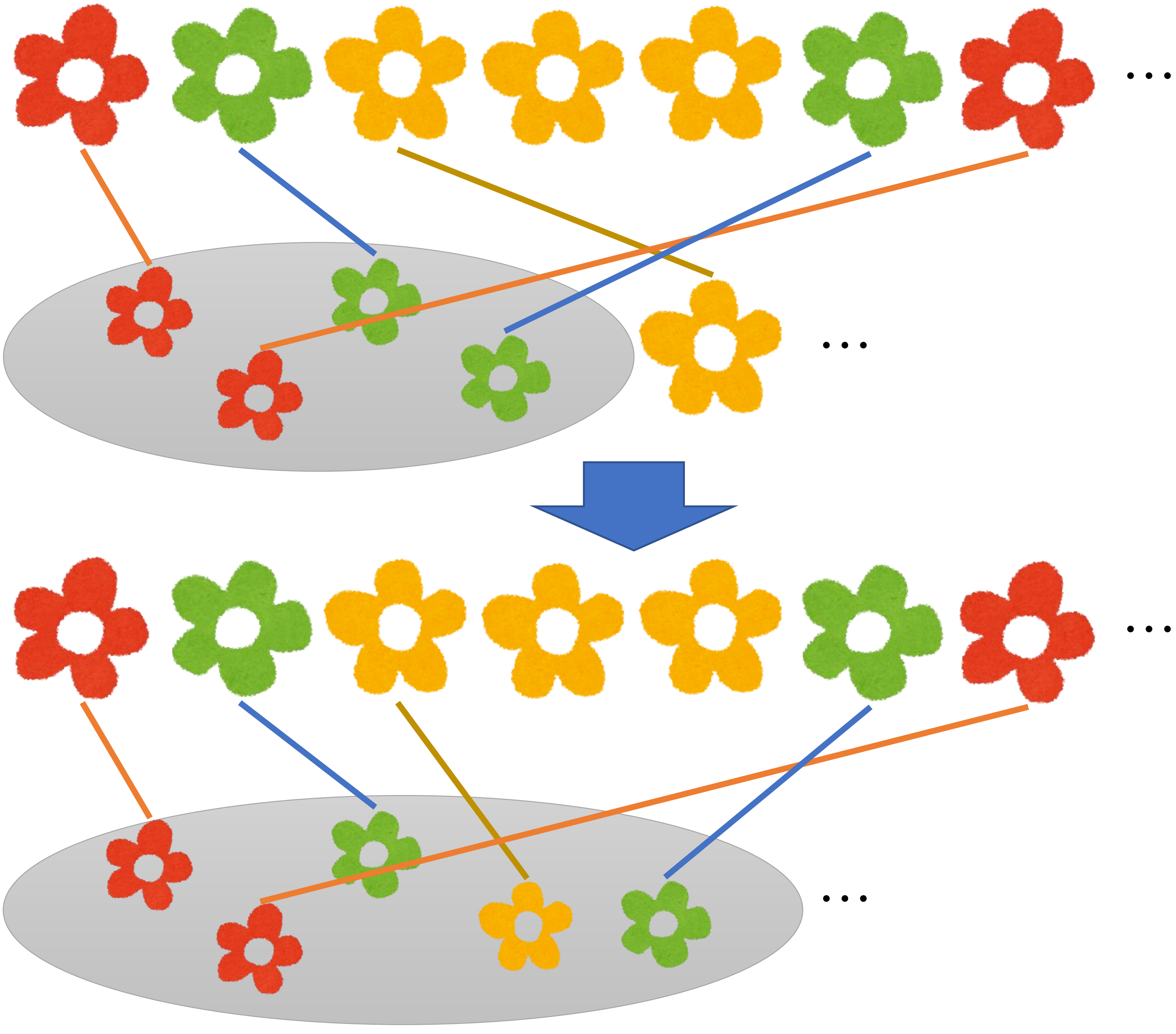


今まで使った各色の個数がわかれば、次のジョイ草がどれと対応するかはわかる



「使った各色個数」がわかれば、この線と交わるほかの線の個数もわかる





- よって、「赤 r 個、緑 g 個、黄 y 個使って、最後の色が c の場合」の、（今まで発生した）交差回数の最小値を動的計画法で計算できる
- 動的計画法の状態数は $O(N^3)$
- 交わる線の個数を求めるのに、毎回 $O(N)$
- よって、計算量は $O(N^4)$

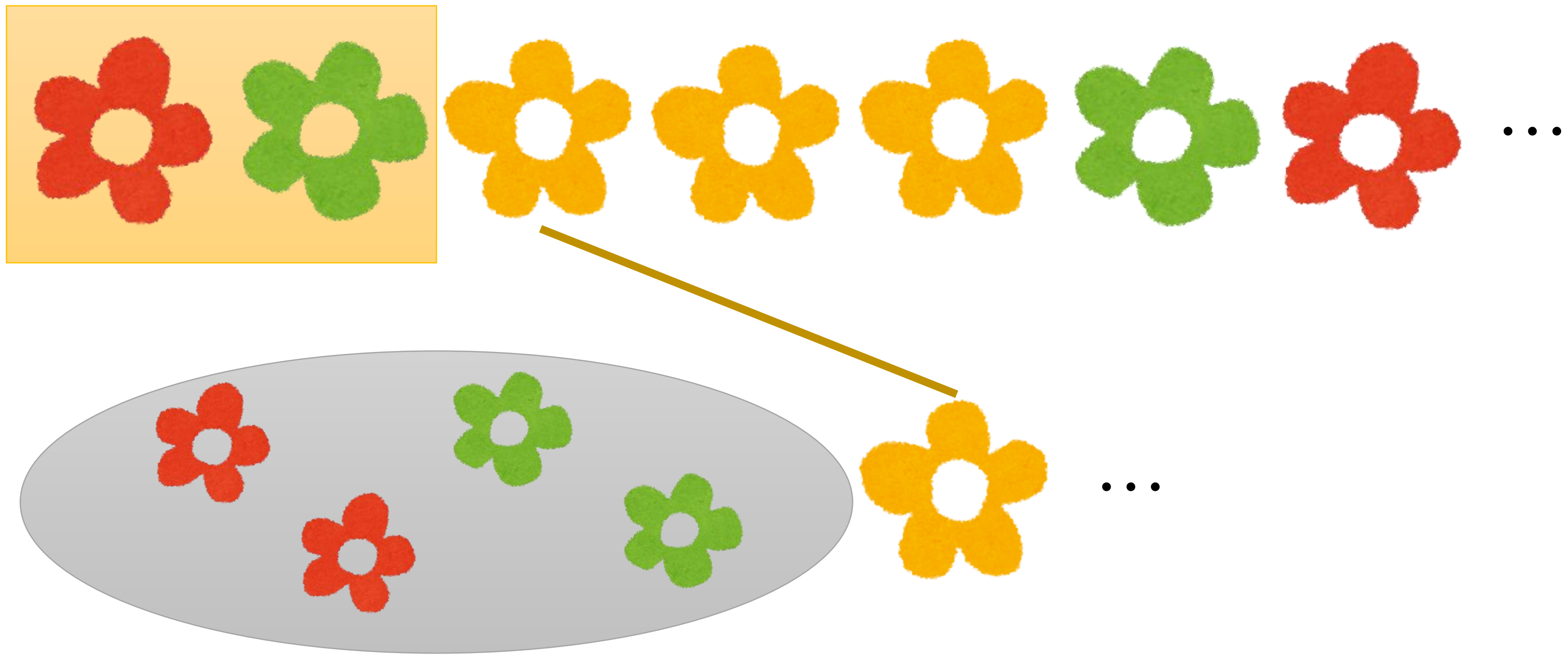
小課題 4 (25 点)

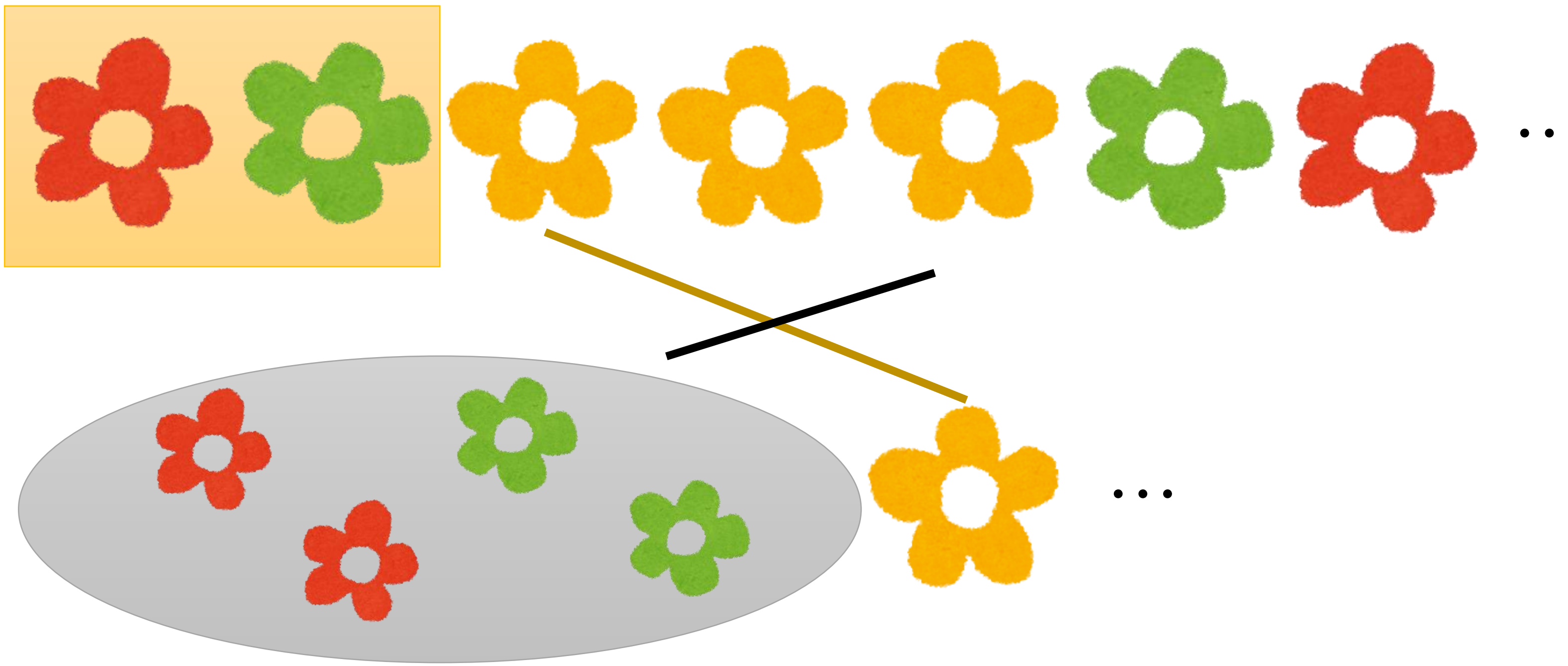
- よって、「赤 r 個、緑 g 個、黄 y 個使って、最後の色が c の場合」の、（今まで発生した）交差回数の最小値を動的計画法で計算できる
- 動的計画法の状態数は $O(N^3)$
- 交わる線の個数を求めるのに、毎回 $O(N)$
- よって、計算量は $O(N^4)$

小課題 4 (25 点)

- よって、「赤 r 個、緑 g 個、黄 y 個使って、最後の色が c の場合」の、（今まで発生した）交差回数の最小値を動的計画法で計算できる
- 動的計画法の状態数は $O(N^3)$
- 交わる線の個数を求めるのに、毎回 $O(N)$
- よって、計算量は $O(N^4)$

これをなんとかしたい





と交わる線の数は、

の中にある赤、緑の個数と、

の中にある赤、緑の個数からわかる

小課題 4 (25 点)

次のものを前計算しておけばよい：

- 各色、各 i について、「i 番目のその色のジョイ草」の位置
- 各位置について、「そこより左にある各色のジョイ草」の数

DP の遷移で、交わる線の数を $O(1)$ で求められる
→$O(N^3)$ 満点が得られる

得点分布