ARC #063 / ABC #047 解説

2016 年 11 月 6 日

A - キャンディーと 2 人の子供

3 個のパック a, b, c を 2 つに分ける方法は

- a と b, c
- b と a, c
- c と a, b

の 3 通り考えればよい (パックには 1 個以上飴が入っているので、パックを 0 対 3 で分けることを考える意味はない)。よって $a = b + c$ か $b = a + c$ か $c = a + b$ が成り立つかどうかを調べて、ひとつでも成り立てば答えは Yes となる。

また、最も飴の個数が多いパック 1 つとそれ以外のパックに分けることだけ考えればよいことに気づくと、a が最大になるよう a, b, c を並び替えたあと $a = b + c$ が成り立つかを調べるという方法も使える。

B - すぬけ君の塗り絵 2 イージー

ひとつの方法は、長方形に対応した $W \times H$ 要素の二次元配列 $A[x][y]$ を用意して、塗りつぶしをシミュレーションすることである。はじめ、全ての要素を 0 などに初期化しておき、黒く塗った場所を 1 に変えていく。最後に残った 0 の個数を数えれば、それがそのまま答えの面積となる。

もうひとつの方法は、長方形内の白い部分は存在すれば長方形になっていることを利用するものである。白い長方形の左下の座標と右上の座標 (最初は $(0, 0)$ と (W, H)) を持っておき、たとえば $x < x_i$ をみたす領域を黒く塗るときは「左下の x 座標が x_i より小さければ x_i に置き換える」とすればよい。

後者の方法のほうが効率よく処理ができるが、この問題の制約下ではどちらの方法を用いても時間的には問題ない。

C - 一次元リバーシ

盤面を、同じ色の石が連続して並ぶ区間に分割して考える。たとえば `BBBWWB` なら $(黒\ 3, 白\ 2, 黒\ 1)$ のように見る。このように見ると、石の並びには白と黒の区間が交互に現れると捉えることが出来る。

石を新たにひとつ打つことで何ができるかを考えると、これは左端か右端にある連続する同じ色の石たちの色をすべて反転させる操作である。白の区間と黒の区間は交互に現れるので、端の区間の色を反転させると、

それは隣の区間とくっついて全体として区間数が 1 減ることになる。

最終的に区間の数を 1 にすることが目的で、石をひとつ打つことで区間数を 1 減らせるから、必要な石の個数は (区間の数) -1 個である。したがって区間の数を求めればよく、これはたとえば隣接する石の色が異なるような箇所がいくつあるかを数えると求められる。

D - 高橋君と見えざる手

まず高橋君が最大の利益を上げるためにどのように行動すればよいかを考える。高橋君が町 i でリンゴを売ったとき、そのリンゴは町 i よりも前にある町で、最もリンゴの値が安いところで買ったとしてよい。つまり、高橋君は町 i でリンゴを売るとき $A_i - \min\{A_1, ..., A_{i-1}\}$ 円の利益をあげられると考えることができる。この値を新たに B_i とおく。すると、ひとつの町で売買するリンゴの数に制限はないので、B_i が最大となるような町でできるだけ多くのリンゴを売ることで収益を最大化することができる。

高橋君の収益を減らすという目的を達成するためには、B_i の最大値を少なくとも 1 減らす必要がある。いま、町 x でリンゴを売るときの収益 B_x がすべての町のなかで最大であるとしよう。B_x は $A_x - \min\{A_1, ..., A_{x-1}\}$ であるが、A_i の値が相異なるという制約からこの式の min の中で最小となるものはただひとつ存在し、$B_x = A_x - A_y$ と書ける。この y を x のペアと呼ぶことにする。B_x の値を減らすためには、A_x を 1 減らすか、ペアである A_y を 1 増やすかのどちらかを行えばよく、これはコスト 1 で行える。

一般に B_i の値が最大となるような町 i は複数個あるかもしれないが、やはり A_i の値が相異なるという制約から、B_x と $B_{x'}$ がともに最大であるとき $x \neq x'$ ならそれぞれのペアである y, y' も異なることがわかる。したがって、最大値が複数個あるときも互いに影響を及ぼすことなく、それぞれコスト 1 で B_x の値を減らしていくことになる。

よって求めるべき答えは B_x が最大となるような x の個数となる。これは A_i を順に見ていき、今まで見た値の最小値を管理しておけば全体で $O(N)$ 時間で求めることができる。

E - 木と整数

元から値 P_r が書かれた頂点 r をひとつ決め、入力の木を r を根とする根付き木であると考える。木のそれぞれの頂点に書き込む整数の偶奇は r からの距離の偶奇によって決まる。以下では、それぞれの頂点に対して、そこに書き込まれる整数の偶奇がどちらでなければならないかという制約が明示的に決められているとして考える。また、元から書かれた値が偶奇の制約に反しているような頂点がある場合は条件を満たすことは不可能なので、そのようなケースはあらかじめ除外してよい。

このとき、頂点 v について、v を根とする部分木に問題の条件を満たすように整数を書き込める必要十分条件を考えよう。

v が葉のとき

v に元々何の値も書かれていないならば、v に書く整数は偶奇の条件さえ満たしていればなんでもよい。これを、「書かれる整数が (偶奇の制約を満たした上で) $(-\infty, \infty)$ の区間に収まっている」のように区間で表現し、単に条件 $(-\infty, \infty)$ と書くことにする。(これより以下では「偶奇の条件を満たした上で」という風にいちいち断らないが、条件について考える時は常に偶奇の条件も満たしている必要がある)

v に値 P_v が書かれている場合は、上の記法に従うと条件 $[P_v, P_v]$ が課されていることになる。

v が子を持つとき

w を v の子とし、w の条件が $[L_w, U_w]$ であるとする。このとき、w 以下にうまく整数を書き込むためには v に書く整数が $[L_w - 1, U_w + 1]$ の範囲になければならない。子が複数いる場合は、それぞれの子についてこのような区間を考え、そのすべての共通部分を $[L_v, U_v]$ として、v に書く整数はこの範囲になければならない。

逆に v に $[L_v, U_v]$ 内の整数を書いた場合、すべての子を根とする部分木についてうまく整数が書き込めるため、これは必要十分条件である。ただし、v に元から値 P_v が書かれていたときは、更に $[P_v, P_v]$ との共通部分をとったものが必要十分条件となる。

以上のように、区間 (と偶奇) によって、うまく整数を書けるための必要十分条件を表すことができる。このような区間は深さ優先探索などをすることで線形時間で求めることができ、最終的に根 r についての条件を表す区間が空でなければ、木全体に問題の条件を満たすように整数を書き込めることがわかる。

それぞれの頂点に対して必要十分条件を表す区間が求まっていれば、具体例をひとつ構成することも容易である。根 r には値 P_r が書かれているので、その子 w には $P_r - 1$ か $P_r + 1$ のうち $[L_w, U_w]$ に収まっている方を書けばよい。同様にして w の子から下の頂点たちにも条件を満たすように整数を書いていくことができる。

F - すぬけ君の塗り絵 2

各点について塗りつぶす方向を左右だけに限定すると、少なくとも周長 $2H + 2$ の長方形が容易に得られる。同様に上下に限定した場合も考えると、答えの下界として $2 \max\{W, H\} + 2$ が得られる。

このことから、答えとなる長方形は直線 $x = W/2$ または $y = H/2$ の少なくとも 1 つとは共通部分を持つことがわかる。よって、ある x_c について長方形が $x = x_c$ と交わる場合だけを考えた問題が解ければ十分である。この問題は分割統治法によって効率よく解くことができる。

分割パート

長方形が $x = x_c$ に交わる場合だけを考えるとき、N 個の点を y 座標が y_c 以下のものと y_c より大きいものに二等分する (y 座標の等しい 2 点はないのでこのような y_c がとれる)。そして、下半分で $x = x_c$ と交わるもの、上半分で $x = x_c$ と交わるものをそれぞれ再帰的に解くと、残りは長方形が (x_c, y_c) を含む場合 (統治パート) となる。

長方形が 1 点 (x_c, y_c) を含む場合だけを解くのにかかる時間を $U(N)$ とすれば、全体でかかる時間 $T(N)$ は $T(N) = 2T(N/2) + U(N)$ となる。

統治パート

まず x 座標が x_c 以下の点たちについて考えよう。最後に残った長方形の左端の x 座標を t だと仮定する。このとき x 座標が t 以下の点たちは左側を塗りつぶしてしまえばよく、x 座標が t より大きい点たちは、(x_c, y_c) を塗りつぶしてはならないので上と下のどちら側を塗りつぶさねばならないかが一意に定まる。つま

り、t を決め打ったとき、長方形の y 座標が $l_t \leq y \leq u_t$ となるような l_t, u_t が決まる。t の候補はたくさんあるように見えるが、l_t, u_t の値が変わるのは点が存在するような x 座標だけなので、そのような場所だけ考えればよい。すなわち、各点について「その点の x 座標を長方形の左端の x 座標としたときの、長方形の y 方向の範囲」を求めればよい。

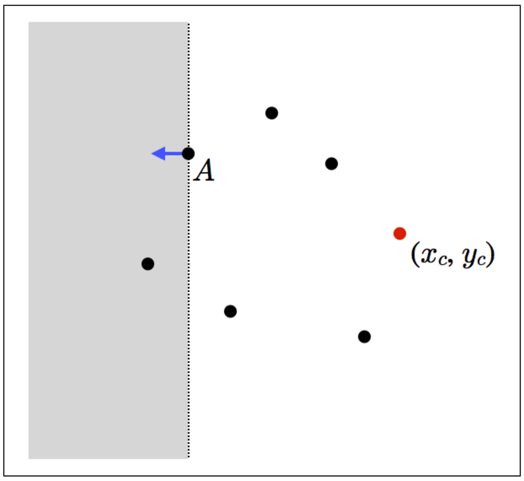

図 1

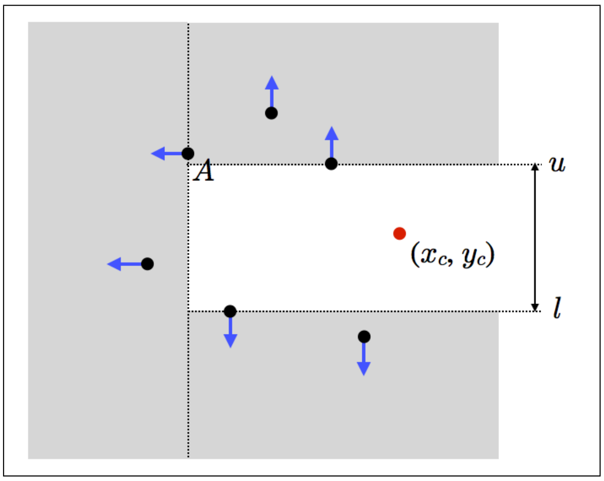

図 2

具体例を図で示す。図 1 の状態で、長方形の左端を点 A が決めると仮定する。すると、図 2 のように残りの点についてもどちらの方向を塗りつぶすかが決まり、残った y 座標の範囲 $[l, u]$ も決まる。

各点についてこのように y 座標の範囲を求めて x 座標とともに記録しておく。すなわち、x 座標と区間のペア $(x, [y_1, y_2])$ を各点に対して計算し、その集合を L とする。これは点を右から見ていけば線形時間で計算できる。

点 (x_c, y_c) の右側にあるような点たちについても同様の計算を行ってその結果を R とすると、解くべき問題は L と R から一つずつ要素をとってきたときの長方形の周長を最大化する問題となる。ここで長方形の周長は「x 座標の差」と「y 座標の区間の共通部分の長さ」の和で求められることに注意せよ。

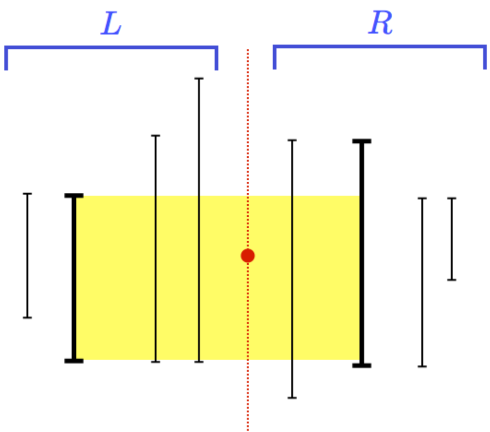

図 3

図 3 に、このように捉えた問題の一例を示す。太く示された区間が選んだ区間であり、黄色い長方形がその区間を選ぶことによってできる長方形である。

この問題は、区間が左と右でそれぞれ単調に広がっていく (あるいは狭まっていく) という事実を用いることで、尺取り法を使って線形時間で解くことができる。具体的には以下のように計算する。

左側の区間 l をひとつ決めたとき、右側の区間としてどれを選ぶか考える。右側の区間は左から順に「l を含むもの」「l を含まないが、l に含まれもしないもの」「l に含まれるもの」の 3 グループに分類することができる。

このうち「l を含むもの」と「l を含まれるもの」の中で最も良い区間は比較的容易に求められる。l を含むものについては、単にその中で最も x 座標の大きいものを選べばよい。l に含まれるものについては、x 座標と区間の長さの和が最も大きいものを選べばよく、この値自体は l そのものに依存しないのであらかじめ最大値を計算しておくことができる。

「l を含まないが、l に含まれもしないもの」は少し難しいが、基本的にはスライド最大値の応用である。このグループに属する区間を両端キュー (deque) などで管理していくが、常に「長方形の周長が降順」となるように不要な区間を適宜取り除いていく。ある区間 I が新たにこのグループに入るときは、deque の末尾の区間による長方形の周長が I によるものより小さい間 deque の末尾の要素を削除して、そのあと追加することにすればよい。こうすると、deque の先頭が常にその時点での最良の区間となって、全体で線形時間を達成できる。

まとめと補足

統治パートが線形時間で解けたので、$T(N) = O(N \log N)$ 時間で元の問題を解くことができる。

以下は補足となるが、最初の下界を使った考察をせず、x 方向にも分割統治法を行って二重の分割統治法でこの問題を解くこともできる。その場合計算時間に $\log N$ の項がひとつ増えることになる。また、統治パートで $O(N \log N)$ 時間の解法を用いることでも $\log N$ の項が増える。このように $O(N \log^2 N)$ 時間の解法も考えられるが、実装によっては通らないことがあるかもしれない。