問題A – プレミアム会員

A - 問題概要

- 過去nヶ月のうちxヶ月間540円，n-xヶ月間525円プレミアム会員費を支払っていた
- nとxが与えられるので合計額を出力しなさい
- $1 \leqq x \leqq n \leqq 100$

A - 解法

- 540x+525(n-x)を計算して出力する
- 標準入力と標準出力の仕方は言語によって異なるので

問題B – ニコニコ文字列

B - 問題概要

- “25”を繰り返した文字列をニコニコ文字列と呼ぶ
- 数字から成る文字列Sにニコニコ文字列となるような部分文字列は何箇所あるか
- 例: 12512525 -> 4箇所(25となるのが3箇所 2525となるのが1箇所)
- 1≦|S|≦100000

B - 部分点解法(30点)

- 二重ループで数え上げる方針
- 外側のforループで始点iを決める
- 内側のforループで始点iから交互に2525…となっている間だけループして5が来る度に答えに1を足す.
  パターンに一致しなくなったら break
- 時間計算量O(N^2) なので 30点が得れる

B - 満点解法 1/2

- ニコニコ文字列となっている部分のみをできるだけ長くなるように残す
  先頭から貪欲に行ってよい.

- 例 125252511525252525
  -> . 252525 . . . 2525 . 25 . 25

- ある連続部分での25が繰り返されている回数をxとすると,
  その連続部分でのニコニコ文字列の切り出し方
  = (“25”の切り出し方)+(“2525”の切り出し方)+…+(2525…25[x回]の切り出し方)
  = x + (x-1) + … + 1
  = x*(x+1)/2 (総和の公式)

B - 満点解法 2/2

- それを全文字列について計算し，足し上げる
- 例:12525251152525525525
  -> . 252525 . . . 2525 . 25 . 25
  答え: 3*4/2 + 2*3/2 + 1*2/2 + 1*2/2 = 11
- 時間計算量 O(N) で満点が得れる
- 最大ケースは25が5万回繰り返されたケースであるが，50000 * 50001 / 2を計算するときに50000 * 50001が32bit符号付き整数型だとオーバーフローするので注意

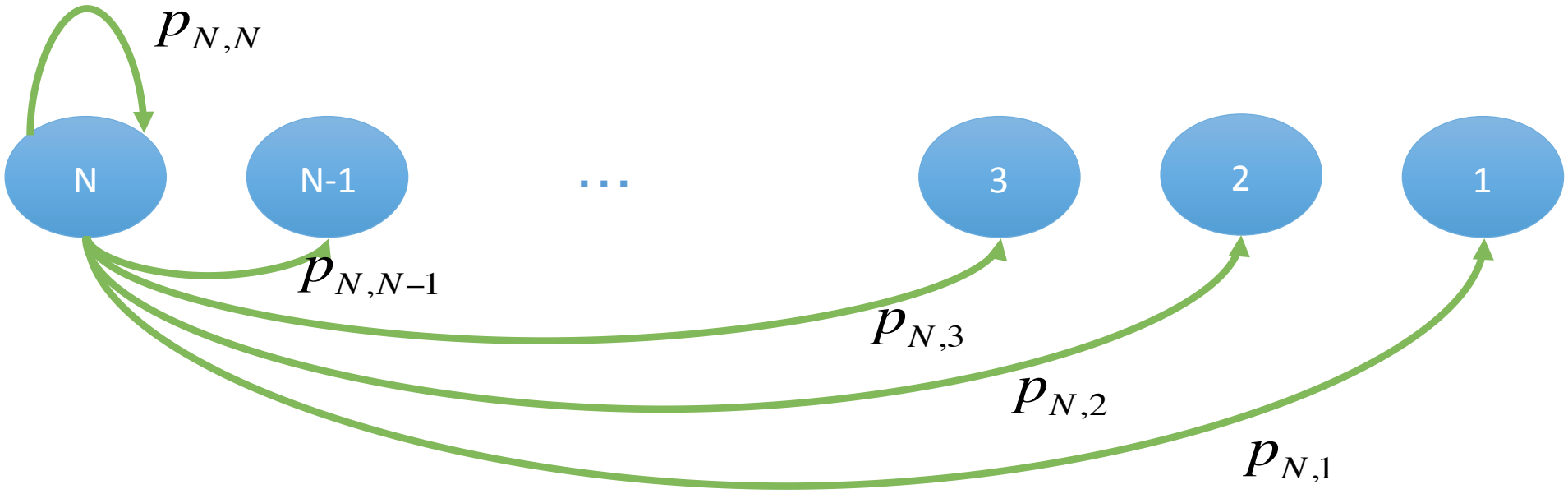
問題C - ゲーマーじゃんけん

概要

- 「少ないもの勝ち」ルールでじゃんけんを行う
  - 1 人の勝者が決まるまでに行うじゃんけんの回数の期待値は？
- この問題は dwango エンジニアからの出題です

解法

- 残り人数を頂点とした状態遷移図を描く
  - 頂点aから頂点bへ遷移する確率を $p_{a,b}$ とおく



解法

- この状態遷移図上で、頂点 N から頂点 1 まで遷移するまでの遷移回数の期待値を求めればよい
- この期待値を $E(N)$ とおくと、次が成り立つ

$$E(N) = \sum_{i=1}^{N} p_{N,i}(E(i)+1)$$

$$E(1) = 0$$

確率 p(N,i) で頂点iに遷移し、
そこから期待値E(i)回で
頂点1に遷移する。
これを全ての頂点について
足し合わせる。

（＊問題文を厳密に解釈すれば N = 1 のときの本問題の答えは
0,0 ではなく 1.0 となるが、計算の都合上無視する）

解法

- この式は両辺に$E(N)$を含むので、移項する

$$E(N) = p_{N,N}(E(N)+1) + \sum_{i=1}^{N-1} p_{N,i}(E(i)+1)$$

$$E(N) = \frac{p_{N,N} + \sum_{i=1}^{N-1} p_{N,i}(E(i)+1)}{1 - p_{N,N}}$$

  - これで$E(1), E(2), \ldots, E(N)$の順に計算できる

p の計算

- N人がランダムに手を出して、グー R 人、チョキ S 人、パー P (= N - R - S) 人となる確率 $f(R,S,P)$ が分かれば、遷移確率 p が計算できる
  - いろいろな方法がある
    - 方法1: 組み合わせ論

$$f(R,S,P) = \frac{{}_N C_R \, {}_{N-R} C_S}{3^N}$$

p の計算

- N人がランダムに手を出して、グー R 人、チョキ S 人、パー P (= N - R - S) 人となる確率 $f(R,S,P)$ が分かれば、遷移確率 p が計算できる
  - いろいろな方法がある
    - 方法2: 組み合わせ論

$$f(R,S,P) = \frac{N!}{3^N (R!)(S!)(P!)}$$

p の計算

- N人がランダムに手を出して、グー R 人、チョキ S 人、パー P (= N - R - S) 人となる確率 $f(R,S,P)$ が分かれば、遷移確率 p が計算できる
  - いろいろな方法がある
    - 方法3: 以下のような漸化式を立てて、動的計画法

$$f(R,S,P)=\frac{f(R-1,S,P)+f(R,S-1,P)+f(R,S,P-1)}{3}$$

計算量

- $O(N^3)$
- スクリプト言語で実装し、実行結果をハードコードする参加者も複数見られた
  - ルール上は問題ない
- E(N) を計算するとき、E(1), E(2), … と求めていくよりメモ化再帰の方が（定数倍の範囲でだが）圧倒的に早い
  - このじゃんけんでは（あいこの場合を除き）ラウンドごとに残り人数が1/3以下になるため、メモ化再帰ならE(N/3+1) からE(N-1)を計算しなくてよい
  - この工夫があれば Ruby でも埋め込まずに解ける

結果

- Total submit: 239
- Total Accept: 67
- First Accept: LayCurse (yukicoder勢) さん
  - 15:29
- 皆様のご参加ありがとうございました

問題D – タクシー

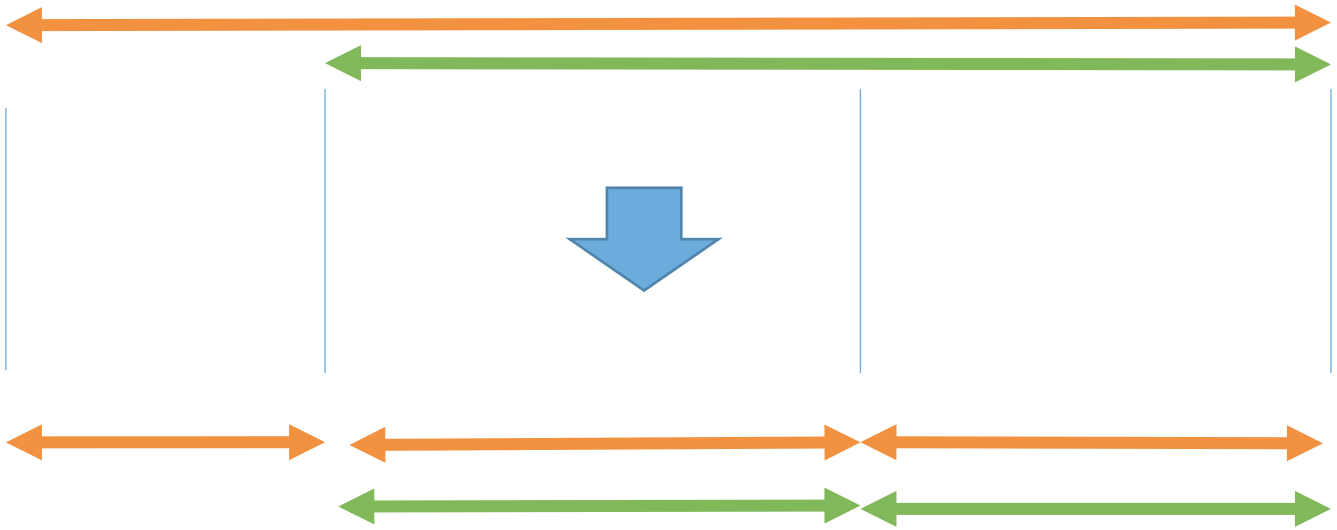
問題概要

- 長さ L の環状の道に N 個の都市があり、各都市ごとに本選出場者とその都市の会場に入る本選出場者数が与えられる。
- 会場に入る人数の合計は、出場者数の合計と一致。
- うまく移動させることで本選出場者数と会場に入る人数が一致するようにしたときの移動距離の合計として考えられる最小値を求めよ。

- $N \leqq L \leqq 10{,}000{,}000$
- $1 \leqq N \leqq 100{,}000$
- $0 \leqq$(各都市に定められた人数)$\leqq 100{,}000$

方針

- 最初に、都市同士の移動を、隣り合う都市同士の移動を複数回繰り返したものとして考えます。
- また、各都市にいる人数を数列で表すことにします。
- 例：4 人、3 人、0 人、1 人　→　[4,3,0,1]



方針

- すると、この問題は、[b1,b2, … , bN] という配列を [c1,c2, … , cN] という形に変換する問題となります。
- 可能な操作は、i 番目の要素から (i+1) 番目の要素に移す操作 (都市 i と都市 (i+1) との間の移動に相当。コストは両都市の距離。) と、1 番目の要素から N 番目の要素に移す操作 (都市 1 と都市 N との間の移動に相当。コストは両都市の距離。) の組み合わせとなります。
- 簡単のため、最初に都市が直線上に並んでいるケース(都市 1 と都市 N との間の移動を無効にした場合)を考えます。

直線の場合の解法

- 例えば、[3,1,2,4] という配列を [4,1,0,5] にしたい場合は、

[3,1,2,4] → [4,0,2,4] (要素 2 から要素 1 に 1 人)

→ [4,1,1,4] (要素 3 から要素 2 に 1 人)

→ [4,1,0,5] (要素 3 から要素 4 に 1 人)

のように左から順に決定していくことができます。

- この決め方で行われる移動が、直線の場合の最適解になります。

※途中で人数が負の数になる場合がありますが、動かす順番をうまく決めることでこの状態を回避できます。

環状の場合の解法

- 最初に、都市 1 と都市 N の間の移動量を定めることで、直線の場合に帰着することができます。
- 移動量として考えられるすべての値を試すことを考えます。
- このとき、参加人数の合計値を超える移動量を試す必要がないことがわかります (例えば、先程の例 [3,1,2,4]→[4,1,0,5] で最初に 11 人の移動をする [14,1,2,-7]→[4,11,2,-7]→[4,1,12,-7]→ [4,1,0,5]は、最初に 10 人の移動をする [13,1,2,-6]→[4,10,2,-6]→[4,1,11,-6]→ [4,1,0,5]という移動よりも多くコストが掛かります。これは、2 通りの移動方法で、2 つともすべて右に数を送っていて、送る量が後者のほうが少ないことから分かります。)

部分点解法 1

- 人数の合計値を S とおくと、最初の移動として両方向に 0 から S までの整数値(都市 N から都市 1 に $-S$ 以上 S 以下の整数値)の移動量を固定して、直線の場合を解くという方針で解を求めることができます。
- 直線の場合を解くアルゴリズムの計算量は O(N) なので、全体で O(NS) の計算量で解くことができます。
- 全体の人数が少ないデータセット 1 をとくことができます。

数列の変形

- さらなる得点を取るために、最初の数列にひと工夫加えます。
- 最初の数列 [b1,b2, … , bN] を、si = b1 + b2 + … + bi (i 番目までの合計)とした数列 [s1,s2, … ,s(N-1)] を考えます。
- 数列 [c1,c2, … , cN] についても同種の変換をした数列 [t1,t2, … ,t(N-1)] を考えます。
- すると、[b1,b2, … , bN] = [c1,c2, … , cN] であることと、[s1,s2, … ,s(N-1)] = [t1,t2, … ,t(N-1)] であることが同値になります。
- よって、s の数列を t の数列に変換する問題に帰着できます。

数列の変形

- 先ほどの数列に対して可能な操作は、i 番目の要素を増減させる操作 (都市 i と都市 (i+1) との間の移動に相当。コストは両都市の距離。) と、数列の全要素を均一に増減させる操作 (都市 1 と都市 N との間の移動に相当。コストは両都市の距離。) の組み合わせとなります。
- すると、数列の全要素を均一に増減させる操作で x だけ増減した場合の、要素 i で行う操作が、

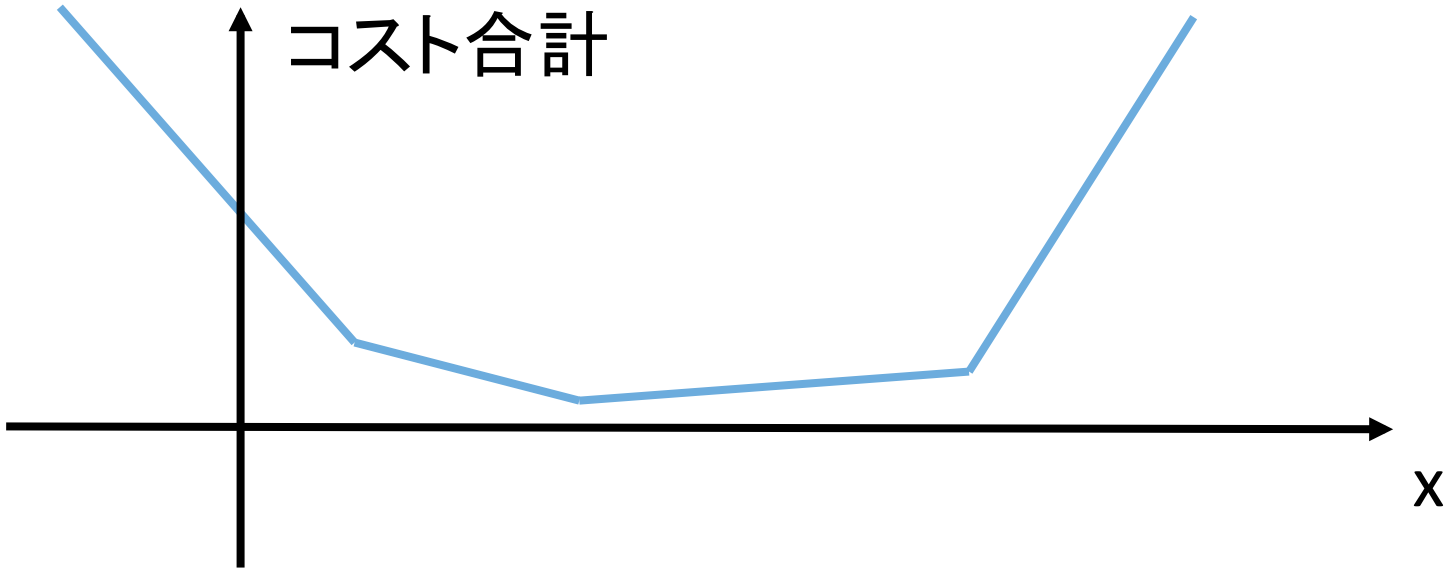
$s_i - t_i + x < 0 \rightarrow -(s_i - t_i + x) *$ (i と i+1 との距離)

$s_i - t_i + x \geqq 0 \rightarrow \quad s_i - t_i + x *$ (i と i+1 との距離)

と $|s_i - t_i + x|$ に比例する関数となります。

関数の遷移

- 先ほどの関数の x の値に対する増減量 f(x) は、下図のようにカクカクした下に凸な関数となります。
- このような関数の最小値を求めるには、すべての頂点(傾きが変わる点)を試せばよく、この傾きはある i で| si – ti + x |=0 となる点です。



部分点解法 2

- すべての要素 i について、x = ti – si として解いてみることにより、先程の各頂点を試すことができます。
- 試した結果のうち最小のものを出力することで答えとなります。

満点解法

- 先程の関数は下に凸な関数です。
- 下に凸な関数の最小値を求めるには、三分探索を用いることで効率的に(O(*N*log(人数の合計)) で)計算することができます。
- 三分探索とは、二分探索の亜種で、区間[a,b] での f(x) の最小値を求める際に、left = (a+a+b)/3, right = (a+b+b)/3 として f(left) と f(right) を比較して、

    f(left) ＜ f(right) → 区間を [left,b] に限定し三分探索

    f(left) ≧ f(right) → 区間を [a,right] に限定し三分探索

と再帰的に実行するアルゴリズムです。

問題E – 電波局

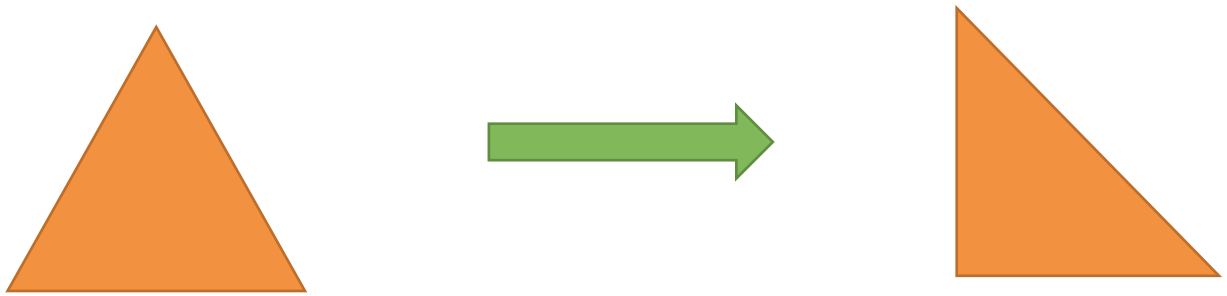
問題概要

- 正三角形状に家が並んだ都市があり、正三角形領域を覆う電波局が M 個あります。
- 「ここに電波局を加えたら新たに覆われるようになる家の数はいくつか」というクエリを Q 個処理してください。

- $1 \leqq N$(都市の辺) $\leqq 1{,}000{,}000{,}000$
- $1 \leqq M \leqq 100{,}000$
- $1 \leqq Q \leqq 100{,}000$

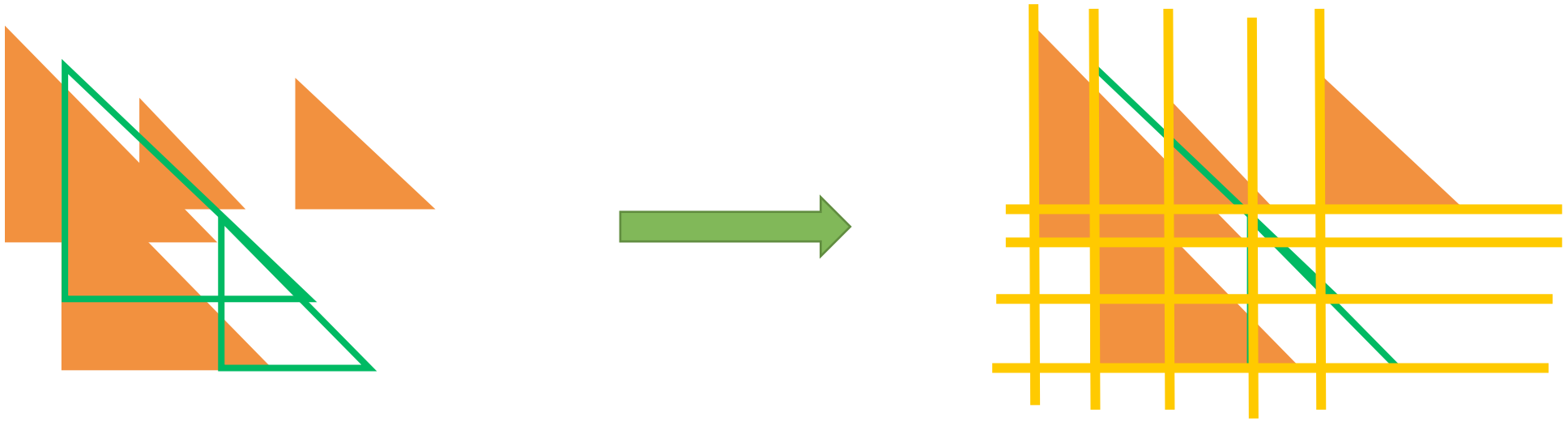
部分点解法 1

- 正三角形を、下図のように直角二等辺三角形と捉えます。
- 家を 2 次元配列上 (初期値は 0) に並べ、各電波局ごとにその電波局が覆う家を表す要素を 1 に変えます。
- 質問クエリも直角二等辺三角形にできて、各質問クエリの電波局が覆う家の要素で 0 の個数を数えるとそのクエリの答えとなります。
- 全体の計算量はO(N^2 (M+Q)) になります。



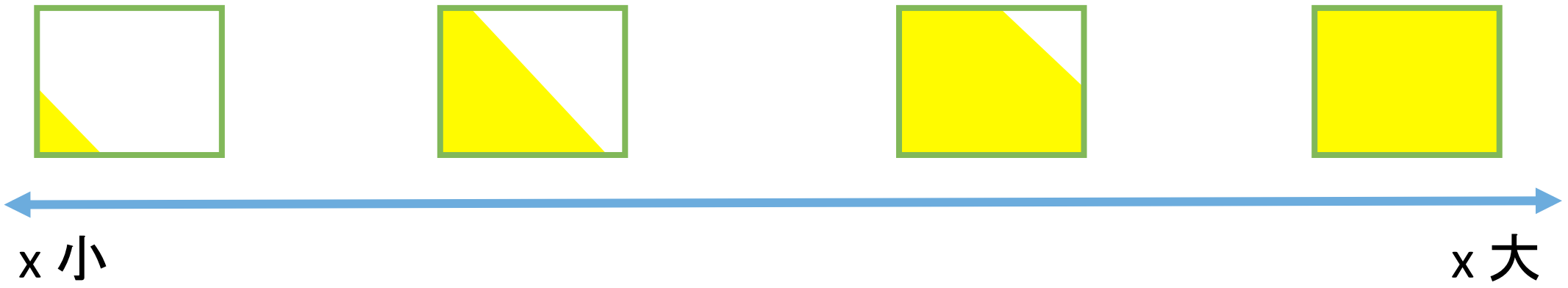
方針

- 今回の場合、家の数がとても多く、1 つ 1 つ処理する方針だと答えが大きいと計算にかなり時間がかかり、制限時間に間に合いません。
- ここで、座標圧縮の方針を適用します。
- 下図のように、初期配置と質問クエリすべての電波局について直角部分で縦横に切った場合を考えます。



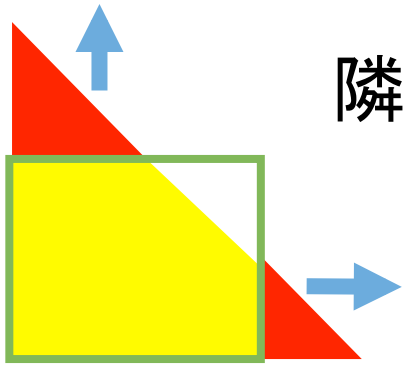
考察

- 各長方形領域について、その領域の中で覆われている家は、ある整数 x を用いて「左下からのマンハッタン距離が x 以下の領域」として表現することができます。



x の計算

- 最初に各正三角形の直角部分を持つ長方形部分に、その直角三角形の大きさと等しい大きさの x を配置します。
- x の大きい順に、その長方形部分の x を確定させます。
- 長方形部分が確定した際、上方向と右方向に新たに別の直角三角形があるものとして、以降はそれらも考慮します。
- 計算は優先度付きキューなどで実現できます。



隣に移動

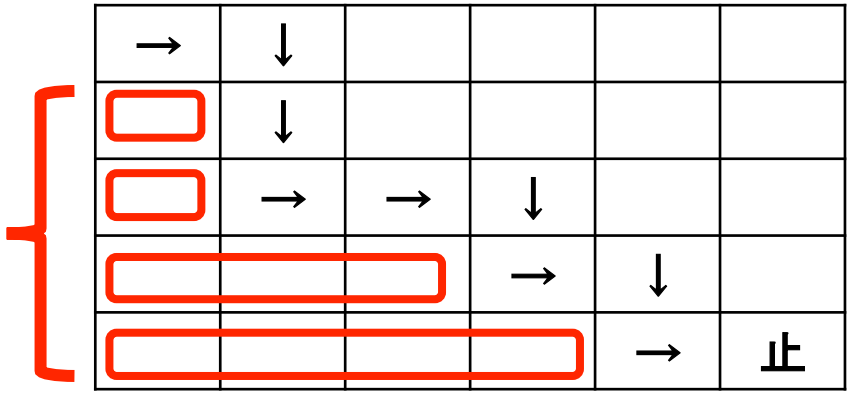
部分点解法 2

- クエリの計算については、各クエリごとにそのクエリが作用しうる長方形部分を 1 個 1 個見る方針で考えます。
- 各長方形部分で、既に置かれている家の領域と新たに置かれる長方形領域(こちらもマンハッタン距離に依存する形になる)を比較し、後者が大きいならその差を足し合わせることで答えとなります。
- 計算量は O($(M+Q)^3$) となります。

満点解法

- クエリ計算部分で、クエリの電波局が長方形部分全体を覆わない箇所は高々 O(幅) 分で、他の覆っている部分はすべて長方形部分全体を覆っていることがわかります。
- 長方形部分全体を覆っている部分については、事前計算と累積和で O(幅) で計算することができます。
- こうすることで O($(M+Q)\uparrow 2 \log (M+Q)$) で計算できます。

この部分を
累積和で
まとめて計算



満点解法

- 実は、先ほどの log を外すことができます。
- x の計算部分で下図のように行を順に処理することで log を外すことができ、O($(M+Q)^2$) にすることができます。