6 感染シミュレーション (Infection Simulation)

解説作成者：米田優峻

小課題 1 (累計 2 点)

制約：$Q \leq 5$，すべての客が時刻 0 に来店し，時刻 10 に退店する。

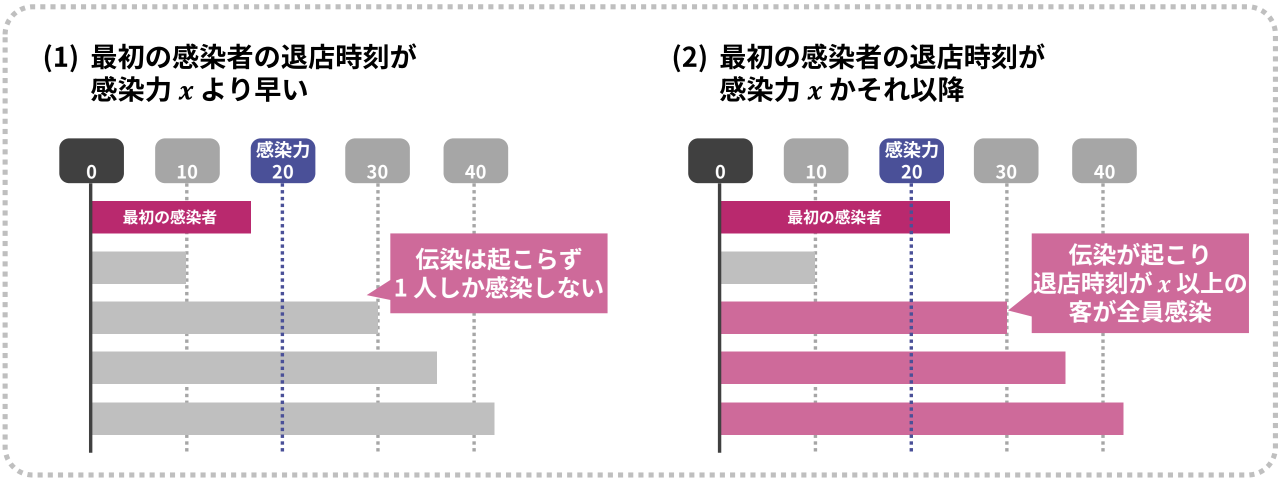
この小課題では，もし感染力 x が 10 以下であれば全員に感染し，そうでなければ最初に感染していた 1 人しか感染しません。したがって，$x \leq 10$ であれば N を，そうでなければ 1 を出力するプログラムを作成すると，正解を出すことができます。

小課題 2 (累計 5 点)

制約：$Q \leq 5$，すべての客が時刻 0 に来店する。

まず，最初に感染した客の退店時刻が感染力 x より小さい場合，伝染が起こる前にその客が帰ってしまいます。そのため，最終的な感染者数は 1 人となります。

そうでない場合，退店時刻が x かそれ以降である客が全員感染します。そのため，最終的な感染者数は $R_i \geq x$ を満たす i $(1 \leq i \leq N)$ の個数となります。単純に for 文を使って個数を計算すると，計算量は $O(NQ)$ となり，$Q \leq 5$ という制約下では実行時間制限に間に合います。



小課題 3 (累計 11 点)

制約：すべての客が時刻 0 に来店する。

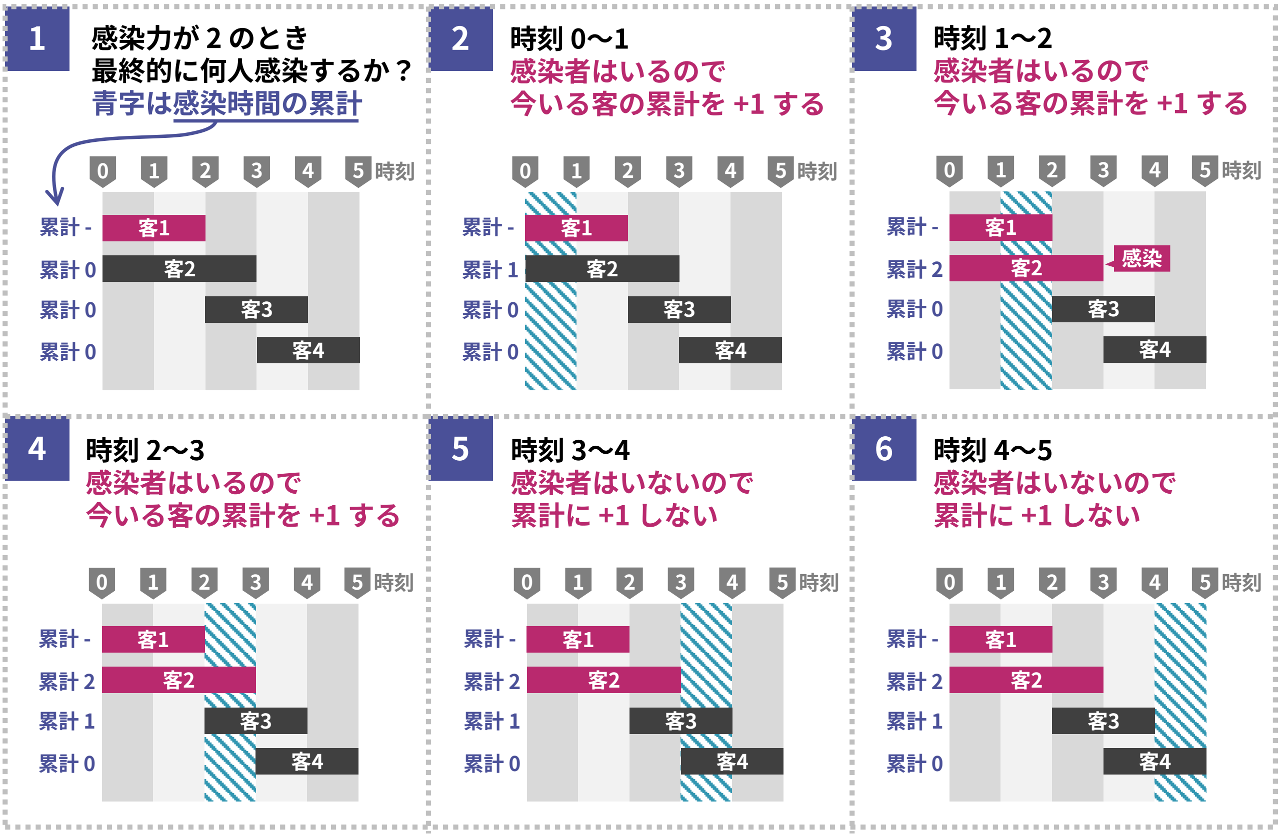
小課題 2 の解法では，$R_i \geq x$ を満たす i の個数を求める際に単純な for 文を使ったため，計算量が $O(NQ)$ となりました。本小課題の制約は $N, Q \leq 100\,000$ であるため，残念ながらこの解法では TLE となります。

しかし，R_i をソートして二分探索をすることを考えてみましょう。すると，計算量が $O((N+Q)\log N)$ に削減され，正解を出すことができます。

小課題 4 (累計 21 点)

制約：$N \leq 500$，$Q \leq 5$，時刻は 500 以下

下図のように，時刻 1 ごとに「感染時間の累計」を更新していくシミュレーションを実装すると，この小課題に正解します。自然に実装した場合，時刻の最大値を T とするとき，計算量は $O(NQT)$ となります。



小課題 5 (累計 32 点)

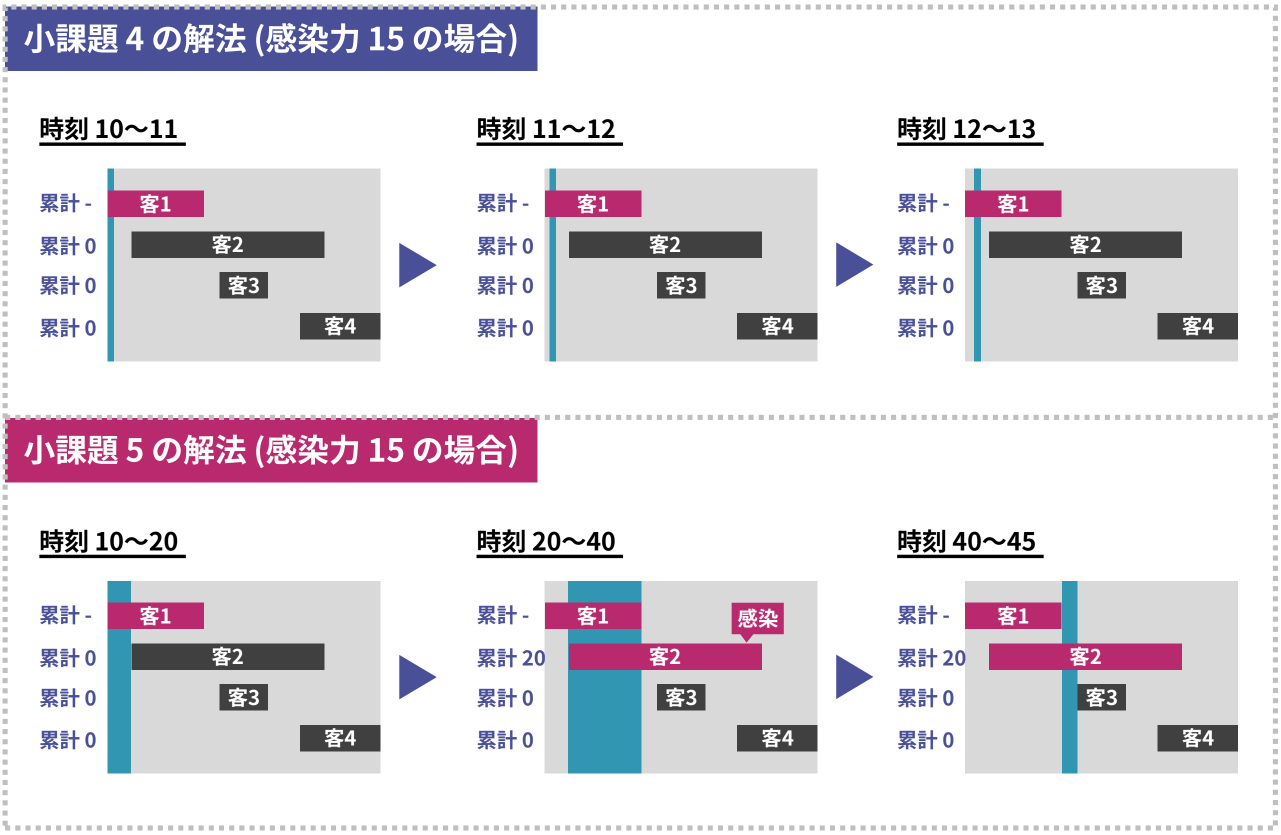
制約：$N \leq 500$，$Q \leq 5$

この小課題では，時刻の最大値が 10^9 まであり得るため，小課題 4 のような「時刻 1 ごとにシミュレーションする解法」は通用しません。

そこで，「来店時刻／退店時刻として出てこない時刻を飛ばしてシミュレーションする」という発想を使うと，計算回数を削減することができます。(このような発想は座標圧縮と呼ばれています)

たとえば客が 4 人おり，それぞれの客が店にいる時刻が 10〜40，20〜80，45〜60，70〜95 であったとします。このとき，小課題 4 の解法では時刻 $10, 11, 12, 13, \cdots, 95$ のように時刻 1 ごとにシミュレーションを行いました。しかし，時刻 $10, 20, 40, 45, 60, 70, 80, 95$ のみを考え，間の時刻を飛ばしてシミュレーションするという方法を使うと，計算回数が削減されます。イメージ図を以下に示します。

なお,「来店時刻／退店時刻として出てくる時刻」としては $2N$ 個あるため, 計算量は $2N \times N \times Q = O(N^2 Q)$ となります。



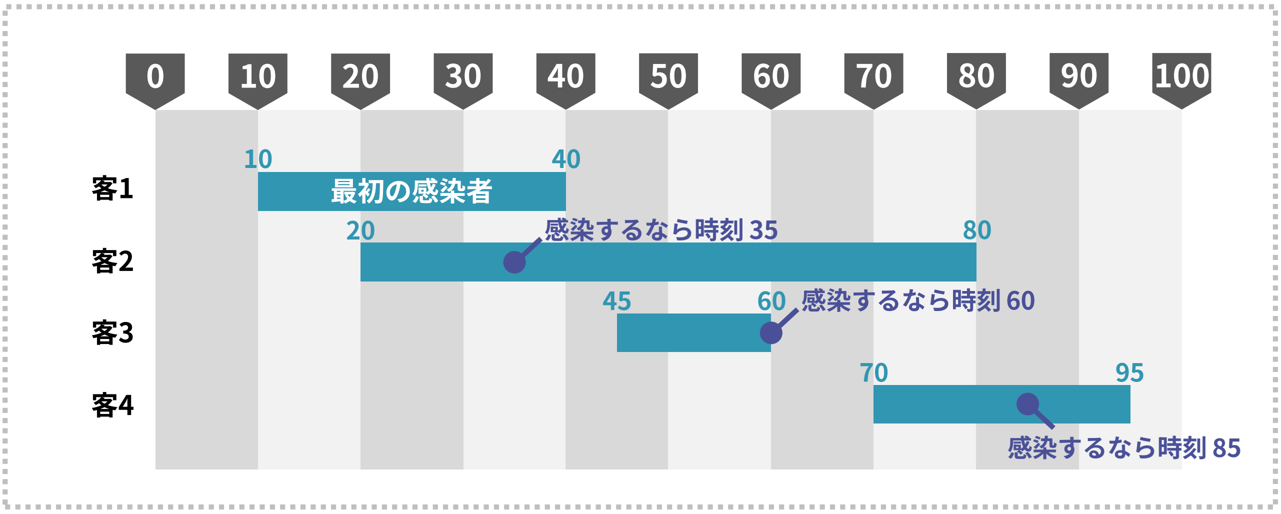
小課題 6 (累計 48 点)

制約：$Q \leq 5$

それでは，先程説明した愚直シミュレーションを何とかして高速化することはできないのでしょうか。まず重要な考察として，**一度感染者がいなくなったら，もう二度と感染は起こりません。**

そのため，感染力が x であるとき，もし客 i が感染するならば，その感染時刻 I_i は必ず (最初の感染者の来店時刻と L_i の大きい方) + x となります。たとえば下図の例で感染力が 15 である場合，$I_2 = 20 + 15 = 35$，$I_3 = 45 + 15 = 60$，$I_4 = 70 + 15 = 85$ となります。

したがって，現在の感染者数を表す変数 `NumInfections` を管理し，来店時刻 L_i，退店時刻 R_i，感染時刻 I_i の $3N$ 個の時刻のみを考えたシミュレーションをすると[*1]，計算量 $O(N \log N)$ で答えを出すことができます。詳しい実装は JOI のホームページに掲載されている実装例を参照してください。



(次ページへ続く)

[*1] 小課題 5 で，来店時刻 L_i，退店時刻 R_i の $2N$ 個の時刻のみを考えたシミュレーションを行ったことを思い出しましょう。

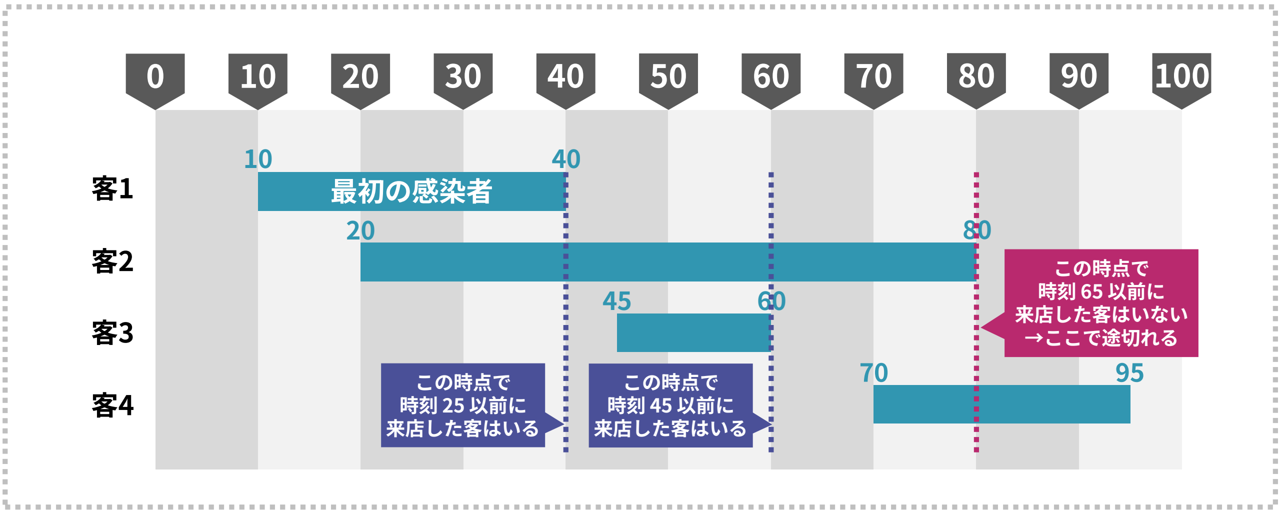
小課題 7, 8 (累計 75 点)

制約：$L_1 \leq L_2 \leq \cdots, L_N, R_1 \leq R_2 \leq \cdots \leq R_N, P_i = 1$。なお，ここでは最初の感染者である客 **1** の来店時刻が一番早いという前提で話を進める。

ここまでは，シミュレーションの高速化という方針で解説を行いました。しかし，ここからは少し考え方を変えて，感染者が店から消える時刻に着目してみましょう[*2]。まず，感染者が店から消える時刻は，以下の条件を満たす時刻 R_j の最小値となります。[*3]

- 客 j の退店時刻 R_j が終わった時点で，次のような客が店にいない。
- 来店時刻が $R_j - x$ かそれより前である。

例として，入力例 1 (下図) の場合を考えてみましょう。まず時刻 $R_1 = 40$ が終わった時点では，来店時刻が $40 - 15 = 25$ かそれより前の客が店にいるので，時刻 40 では感染者が消えません。続いて時刻 $R_2 = 60$ が終わった時点でも，来店時刻が $60 - 15 = 45$ かそれより前の客が店にいるので，時刻 60 では感染者が消えません。しかし時刻 $R_3 = 80$ が終わった時点では，来店時刻が $80 - 15 = 65$ かそれより前の客が店にいません。よって時刻 80 に店から感染者が消えます。



[*2] 小課題 6 で，一度感染者が店にいなくなったら，もう二度と感染が起こらないという考察がありました。そのため，感染者が消える時刻が計算できれば，この問題はほぼ解けたも同然です。
[*3] もちろん，「最初の感染者の退店時刻かそれより後」の中での最小値を指しています。

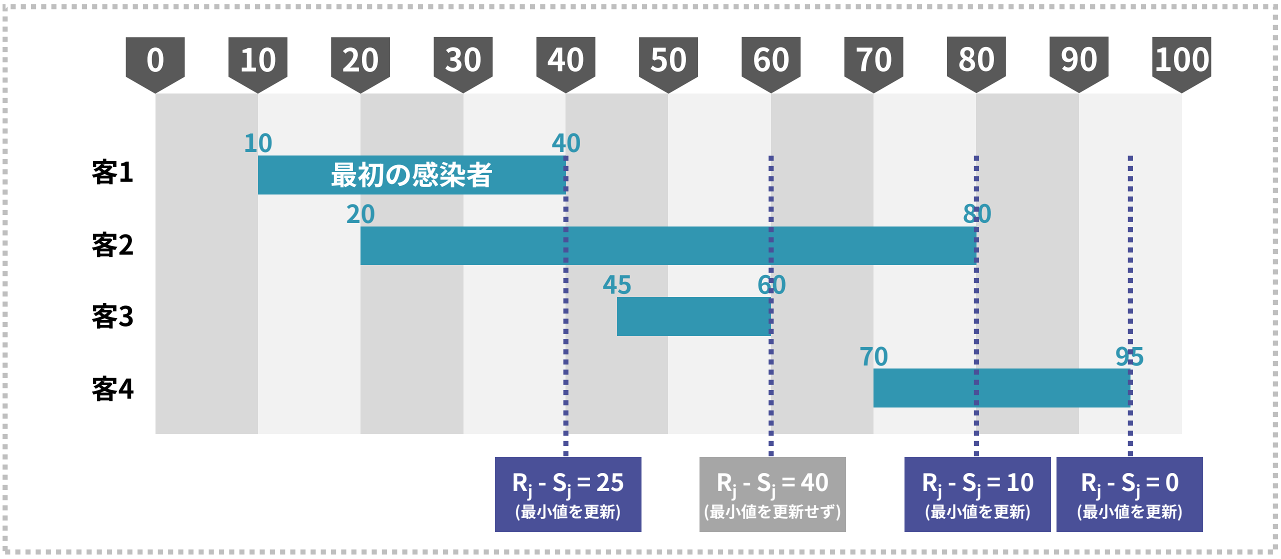
したがって，各 $j = 1, 2, \cdots, N$ について，時刻 R_j が終わった時点で店にいる客における来店時刻の最小値をを S_j とするとき，感染力が x のシナリオにおける「感染者が消える時刻」は次式で表されます。

- $R_j - S_j < x$ となる最小の R_j

そこで $S_1, S_2, \cdots, S_N$ の値は，ソートを利用することで計算量 $O(N \log N)$ で求めることができます。また，$R_j - S_j < x$ となる最小の x の値についても，

- R_j の小さい順に見ていったときに $R_j - S_j$ の最小値を更新する位置

を前計算しておけば，二分探索によって計算量 $O(\log N)$ で求めることができます (下図参照)。以上で「感染者が消える時刻」を高速に計算することが出来ました。



最後に，本題である「何人感染するかの計算」に戻ります。感染力が x，感染が止まる時刻を t とするとき，客 i が感染する条件は以下の式で表されます。

- 来店時刻 L_i が $t - x$ かそれ以前
- 店にいた時間 $R_i - L_i$ が x 以上である[*4]

したがって，感染者数の計算は「二次元平面上の座標 $(L_1, R_1 - L_1), \cdots, (L_N, R_N - L_N)$ に点があるとき，ある長方形領域に何個の点があるかを求める」という典型問題と等価です。そしてこの典型問題は平面走査 + セグメント木により $O((N + Q) \log N)$ 時間で解けるため，小課題 8 に正解することができます。

[*4] もちろん，最初の感染者である客 1 の来店時刻が一番早いとは限らない場合，小課題 9 の解説に書かれている通り，さらに「退店時刻 R_i が (客 1 の来店時刻) + x かそれ以降」という条件が必要になります。しかし，本小課題では $P_i = 1$ であり，客 1 が一番早く来る仮定をおいても良いため，この条件を考える必要はありません。

小課題 9, 10 (累計 100 点)

制約：追加の制約はない

まずは「感染者が消える時刻」について考えましょう。小課題 8 と同様に各客 j $(1 \leq j \leq N)$ に対して S_j を定義したとき，感染者が消える時刻は次のようになります。

- $R_j \geq$ (最初の感染者の退店時刻) を満たす中における，$R_j - S_j < x$ となる最小の R_j

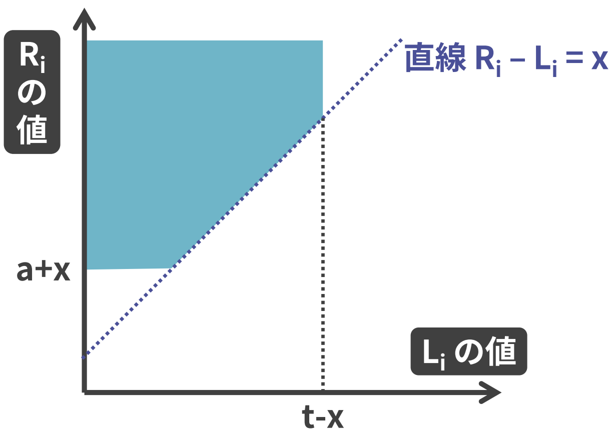
小課題 8 と比べて「$R_j \geq$ (最初の感染者の退店時刻)」という条件が新たに追加されているため，小課題 8 のように最小値の更新位置を前計算する方法では解くことができません。しかし，RMQ[*5] 上で二分探索をすると，各シナリオについて $O(\log^2 N)$ 時間で答えを出すことができます。

＊＊＊

問題は何人が感染するかです。感染力が x，感染が止まる時刻を t，最初の感染者の来店時刻を a とするとき，客 i が感染する条件は以下の式で表されます。小課題 8 と比べて 2 つ目の条件が増えています。

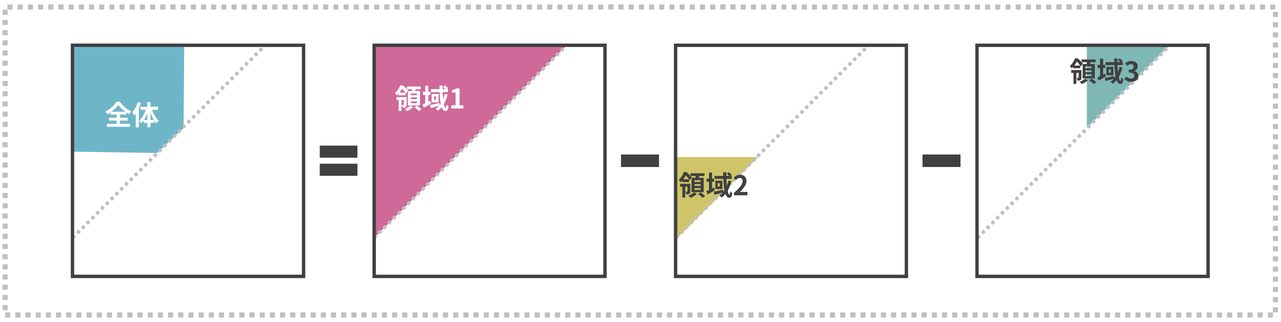
- 来店時刻 L_i が $t - x$ かそれ以前
- 退店時刻 R_i が $a + x$ かそれ以降
- 店にいた時間 $R_i - L_i$ が x 以上である

したがって，二次元平面上に点 $(L_1, R_1), (L_2, R_2), \cdots, (L_N, R_N)$ をプロットしたとき，感染者数は下図のような領域に含まれる点の数と同じになります。



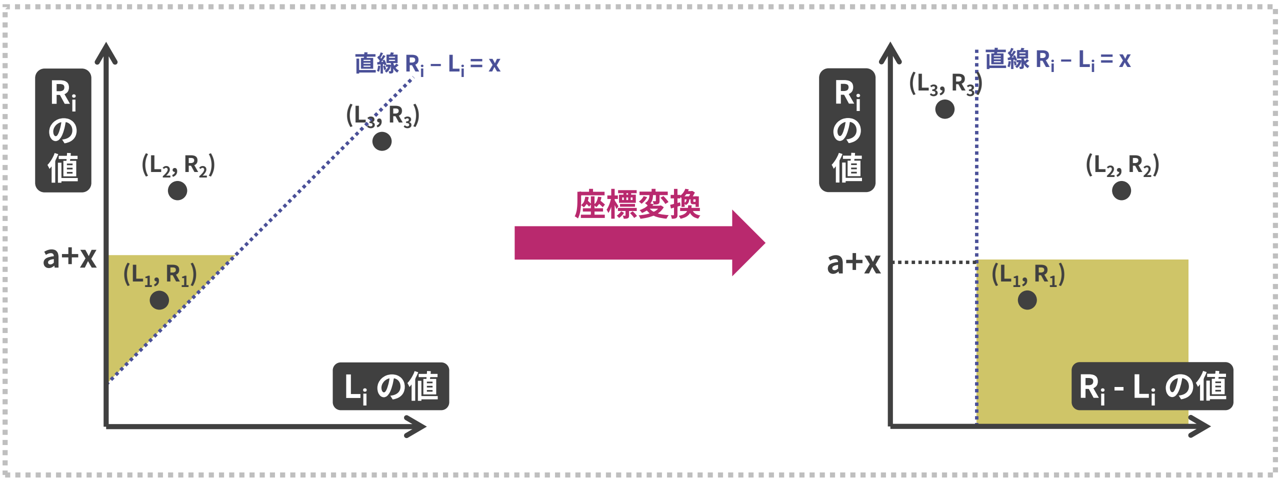
*5 Range Minimum Query の略。セグメント木の一種。知らない方は Google 検索を試してみよ。

それでは，このような点の数はどうやって求めれば良いのでしょうか。まず，全体の領域は下図の領域 (1), (2), (3) を使って，(1) - (2) - (3) という形で表すことができます。また，領域 (1) にある点の数は，$R_i - L_i$ を小さい順にソートして二分探索することで簡単に計算できます。



問題は領域 (2) にある点の数ですが，下図のように横軸を $R_i - L_i$，縦軸を R_i とした座標系に変換すると，長方形領域内に何個の点が含まれるかを求める問題に帰着されます。したがって，小課題 8 のように平面走査 + セグメント木を使うと，各シナリオにつき $O(\log N)$ で計算できます。領域 (3) も同様です。

以上の考察を使うと，各シナリオにつき計算量 $O(\log N)$ で感染者数を計算することができます。これでようやく，最後の小課題を解くことができました。



総評

この問題はかなり多くの考察ステップを必要とする難易度の高い問題です。これを解かれた方はヨーロッパ女子情報オリンピック (EGOI) で優勝を狙える実力がありますので，ぜひ自信を持っていただければと思います。