Xmas Contest 2010

Solutions

Problem A: Christmas Trees

くまの家ときつねの家の距離は $c + d$ Metr (間にねこの家がある場合) または $|c - d|$ Metr (そうでない場合) である．ただし, 家が異なる場所にあるという条件により, $c = d$ のときは必ず $c + d$ Metr になることに注意しなければならない.

x を整数として, x Metr の区間にあるクリスマスツリーの本数の最小値と最大値を考える．必要ならば a と b を入れ替えて $a \le b$ としてよい. x を $a + b$ で割った商を q, 余りを r とする.

最小値について．適切に平行移動することで, 区間の左端から左に 1/2 Metr の位置にクリスマスツリーがあるような最適解が存在することがわかる．そこから右にクリスマスツリーの間隔が $b, a, b, a, \ldots$ となっている場合が最小であり, $0 \le r < b$ のとき $2q$ 本, $b \le r < a + b$ のとき $2q + 1$ 本となる.

最大値について．適切に平行移動することで, 区間の左端ちょうどの位置にクリスマスツリーがあるような最適解が存在することがわかる．そこから右にクリスマスツリーの間隔が $a, b, a, b, \ldots$ となっている場合が最大であり, $0 \le r < a$ のとき $2q + 1$ 本, $a \le r < a + b$ のとき $2q + 2$ 本となる.

Problem B: Simple Parsing

丁寧に構文解析を行ってもよいが, 数式中に現れる整数のうち奇数のものが偶数個あるか奇数個あるかによって答えが定まることに着目すると, (‘1’, ‘3’, ‘5’, ‘7’, ‘9’ のどれか)(数字以外または終端) となっている箇所を数えればよいことがわかる.

Problem C: Connect The Decoration

無向グラフとその頂点の部分集合が与えられたときに最小 Steiner 木を求める問題, と言い換えることができる. すなわち, 元のグラフの部分グラフであって, 少なくとも指定された頂点はすべて含むような木のうち, コストが最小のものを求める問題である.

最小 Steiner 木を求める問題は一般には NP 完全であるが, この問題では使わなくてもよい頂点数 (K とおく) が 10 以下と少ないことを利用する. 使わなくてもよい頂点のうちどの頂点を使うかは 2^K 通りあり, これらをすべて試す.

すると, それぞれの場合に対して最小全域木問題 (頂点集合に対してそれらを繋ぐコスト最小の木を求める) を解けばよい. 最小全域木は Kruskal 法や Prim 法を用いて $O(M \log N)$ 時間で求められ, 問題全体が $O(2^K(N+M)\log N)$ 時間で解けることになる. この方法ではいくつかのテストケースに対しては時間超過となる可能性がある.

Kruskal 法における計算量のボトルネックはグラフの辺をコストによりソートするステップであるが, 最初に辺をソートしておくことによって, 計算量を $O(M \log M + 2^K M\alpha(M,N))$ に減らすことができる (α は Ackermann の逆関数).

Problem D: Presents

この問題では, 買うプレゼントの個数に制約がないが, 疲れ度がプレゼントの個数の 2 乗に比例して増えるため, ある程度以上多くプレゼントを買うのは無駄であるとわかる.

そこでまず, 買うプレゼントが高々 M 個であるとして問題を解くことを考える. すると, $0 \le i \le N$, $0 \le j \le M$ なる各 i, j に対して,「店 i までの買い物を終えて店 $i+1$ に着くまでの, プレゼントを合計 j 個買った場合における, (かかった金額) + (たまった疲れ度) の最小値」を順次求めていけばよい. この値を $c_{i,j}$ と表すと, 最終的に求めたい値は $c_{N,j} - Qj$ の $P \le j \le M$ における最小値である.

$c_{i,j}$ を求める単純な方法として, 以下の手順が考えられる.

for $i := 0$ **to** N:
  for $j := 0$ **to** M:
    $c_{i,j} := +\infty$.
$c_{0,0} := 0$.
for $i := 0$ **to** $N-1$:
  for $j := 0$ **to** M:
    /* 店 $i+1$ に寄らない場合 */
    $c_{i+1,j} := \min\{c_{i+1,j},\ c_{i,j} + j^2(d_{i+2} - d_{i+1})\}$.
    for $k := j+1$ **to** M:
      /* 店 $i+1$ で $k-j$ 個のプレゼントを買う場合 */
      $c_{i+1,k} := \min\{c_{i+1,k},\ c_{i,j} + (a_{i+1}(k-j) + b_{i+1}) + k^2(d_{i+2} - d_{i+1})\}$.

ただし $d_{N+1} = D$ とする. この方法の時間計算量は $O(NM^2)$ である. 値段が買うプレゼントの個数の 1 次式であることを利用すると, 以下のように改良することができる.

for $j := 1$ **to** M:
  $c_{0,j} := +\infty$.
$c_{0,0} := 0$.
for $i := 0$ **to** $N-1$:
  for $j := 0$ **to** M:
    $t_j := c_{i,j} + b_{i+1}$.
  for $j := 0$ **to** $M-1$:
    $t_{j+1} := \min\{t_{j+1},\ t_j + a_{i+1}\}$.
  for $j := 0$ **to** M:
    $c_{i+1,j} := \min\{c_{i,j},\ t_j\} + j^2(d_{i+2} - d_{i+1})$.

ここで, 補助的な配列 (t_j) によって, 店 $i+1$ でプレゼントを 0 個買う場合でも b_{i+1} 円かかるとして計算した場合の, 店 $i+1$ での買い物を終えるまでの最小コストを求めている. この方法の時間計算量は $O(NM)$ である. また, 配列 $(c_{i,j})$ は使い回しができるために, 空間計算量 $O(M)$ で処理できる.

さて, 必要な M の上界を与えよう. $M \ge P$ とする. プレゼント $M+1$ 個以上買うことが無駄になるための十分条件を考える. $M+1$ 個以上の買い方に対して, 最後の 1 個を買わない場合と比較すると, かかった金額は増加し, 疲れ度は少なくとも $(M+1)^2 - M^2 = 2M+1$ 増加する (最後の 1 個を買った後に 1

Metr 以上進むため). よって, $2M+1 \geq Q$ であれば $M+1$ 個以上買う場合を考えなくてよい. すなわち, $M = \max\{P, \lceil (Q-1)/2 \rceil\}$ ととることができ, $O(N \max\{P, Q\})$ の解法を得る.

さらなる高速化のためには, M を固定しないというアイデアが必要である. 先程の考察と同様に, 店 i にいる段階でプレゼントを $m_i + 1$ 個以上買うのが無駄になるための十分条件を考える ($m_i \geq P$). 店 i までに買った最後の 1 個を買わない場合と比較すると, 疲れ度は少なくとも $(2m_i+1)(D-d_i)$ 増加する. よって, $m_i = \max\{P, \lceil (Q/(D-d_i)-1)/2 \rceil\}$ ととれて, 店 i までの段階ではこの個数までを考えれば十分であり. その他の範囲では $c_{i,j} = \infty$ としても同じ結果が得られる. この考察から, 以下の解法を得る.

$M := \max\{P, \lceil (Q/(D-d_N)-1)/2 \rceil\}$.
for $j := 1$ **to** M:
    $c_j := +\infty$.
$c_0 := 0$.
for $i := 0$ **to** $N-1$:
    $m := \max\{P, \lceil (Q/(D-d_i)-1)/2 \rceil\}$.
    for $j := 0$ **to** m:
        $t_j := c_j + b_{i+1}$.
    for $j := 0$ **to** $m-1$:
        $t_{j+1} := \min\{t_{j+1}, t_j + a_{i+1}\}$.
    for $j := 0$ **to** m:
        $c_j := \min\{c_j, t_j\} + j^2(d_{i+2} - d_{i+1})$.
$answer := +\infty$.
for $j := P$ **to** M:
    $answer := \min\{answer, c_j - Qj\}$.
print $answer$.

この方法の計算量を分析しよう. $d_1 < \cdots < d_N < d_{N+1} = D$ が重要である.

$$
\begin{aligned}
\sum_{i=1}^{N} m_i &= \sum_{i=1}^{N} \max\left\{P, \frac{1}{2}\left\lceil \frac{Q}{D-d_i} - 1 \right\rceil\right\} \\
&\leq \sum_{i=1}^{N} P + \sum_{i=1}^{N} \frac{1}{2}\left\lceil \frac{Q}{D-d_i} - 1 \right\rceil \\
&\leq NP + \frac{Q}{2}\sum_{i=1}^{N} \frac{1}{D-d_i} \\
&\leq NP + \frac{Q}{2}\sum_{i=1}^{N} \frac{1}{N+1-i} \\
&= NP + \frac{Q}{2}\sum_{i'=1}^{N} \frac{1}{i'} \\
&= O(NP + Q\log N)
\end{aligned}
$$

であるから, 十分に高速に動作することがわかる.

Problem E: Donuts

計算誤差の範囲で正確に解く方法と近似解法を紹介する.
まず, 正確な解法は次のような流れである.

(1) 楕円の方程式の整理
(2) 「興味がある」X 座標の列挙
(3) 各 X 座標の間に存在する楕円弧の列挙
(4) 面積の計算

以下に具体的な方法を示す.

(1) 楕円の方程式の整理

回転前の楕円は $\begin{pmatrix} x & y \end{pmatrix} \begin{pmatrix} 1/a^2 & 0 \\ 0 & 1/b^2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 1$ と書けるので, θ 回転により,

$$\begin{pmatrix} X & Y \end{pmatrix} \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} 1/a^2 & 0 \\ 0 & 1/b^2 \end{pmatrix} \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} X \\ Y \end{pmatrix} = 1$$

$$\begin{pmatrix} X & Y \end{pmatrix} \begin{pmatrix} (1/a^2)\cos^2\theta + (1/b^2)\sin^2\theta & (1/a^2 - 1/b^2)\cos\theta\sin\theta \\ (1/a^2 - 1/b^2)\cos\theta\sin\theta & (1/a^2)\sin^2\theta + (1/b^2)\cos^2\theta \end{pmatrix} \begin{pmatrix} X \\ Y \end{pmatrix} = 1$$

となる. 中央の行列を $\begin{pmatrix} A & B \\ B & C \end{pmatrix}$ とおき, 以下楕円は $AX^2 + 2BXY + CY^2 = 1$ の形で扱う. ここで, $A > 0$, $C > 0$, $AC - B^2 > 0$ が成り立つことがわかる. また, κ 倍拡大は各係数を $1/\kappa^2$ 倍すればよい.

(2) 「興味がある」X 座標の列挙

いずれかの楕円の左右の端となっている X 座標, 異なる 2 楕円の交点の X 座標をすべて列挙しソートする. これらの X 座標は高々 $8N^2$ 個である.

楕円 $AX^2 + 2BXY + CY^2 = 1$ の左右の端は, $X = \pm\sqrt{C/(AC - B^2)}$ である.

2 楕円 $A_1X^2 + 2B_1XY + C_1Y^2 = 1$, $A_2X^2 + 2B_2XY + C_2Y^2 = 1$ の交点は, 斉次方程式 $(A_1 - A_2)X^2 + 2(B_1 - B_2)XY + (C_1 - C_2)Y^2 = 0$ を解いて 1 文字を消去することで計算できる.

(3) 各 X 座標の間に存在する楕円弧の列挙

(2) の手順で得られた $X_1, X_2, \ldots$ に対して, 各 $X_i \le X \le X_{i+1}$ の範囲に存在する楕円弧を列挙し, Y 座標の順番でソートする.

$X_i \le X \le X_{i+1}$ の範囲で楕円弧たちの上下の位置関係は変わらない. よって, 各楕円と直線 $X = (X_i + X_{i+1})/2$ が交わるかどうか, 交わるならば交点の Y 座標を調べればよい. 交わるときは 2 点で交わるが, 上側の楕円弧と下側の楕円弧は区別しておく.

なお, (2) の列挙時に, 計算誤差の影響で同じ X 座標の値が異なる値として処理されてしまうことがある. しかし, $X_i \le X \le X_{i+1}$ の範囲の面積への寄与は高々 $200(X_{i+1} - X_i)$ であるから, 幅が十分小さい範囲は適切に見積もった上で無視することができる.

(4) 面積の計算

各 $X_i \le X \le X_{i+1}$ の範囲で, (3) の手順でソートされた楕円弧を Y 座標の順番に見ていく. ドーナツの外側の楕円については下側で $+1$, 上側で -1, ドーナツの内側の楕円については下側で -1, 上側で $+1$, という数え方をすると, 値が N となっている間ですべてのドーナツが重なる.

楕円の方程式を Y について解いた $Y = (-B \pm \sqrt{C - (AC - B^2)X^2})/C$ を積分すればよい. あるいは, 縦横の拡大によって円での求積に帰着させて考えることもできる.

ここでは X 座標で区切って面積を求める方法を紹介したが, すべての楕円は原点を中心とするので, 極座標を考えて θ 座標で区切っていってもよい. その場合, 楕円の端を考慮する必要がない.

次に, 近似解法を考える.

直交座標である X 座標上で面積を求めたい部分の長さが $\ell(X)$ であるとき, 全体での面積は $\int_{-\infty}^{+\infty} \ell(X)dX$ となる. あるいは極座標では, ある θ 座標上で面積を求めたい部分が区間 $[r_{\min}(\theta), r_{\max}(\theta)]$ であるとき (原点から伸びる半直線上で, すべてのドーナツが重なる部分は繋がった区間である), 全体での面積は $\int_0^{2\pi} \frac{1}{2}(r_{\max}(\theta)^2 - r_{\min}(\theta)^2)d\theta$ となる.

正確な解法における X 座標あるいは θ 座標を列挙する手順を省略し, 適当な間隔で X 座標あるいは θ 座標をたくさんとり, 数値積分を行うことにより近似的に面積が求まる. 実行時間を抑えつつ誤差を減らす工夫が必要となる.

数値積分を行うにあたっては, 台形近似では足りず, Simpson の公式 (2 次式での近似) が適している. さらに, $\ell(X)$ は微分係数が発散してしまう点があり誤差が大きくなってしまうので, 極座標による計算が適切である.

Problem F: Christmas Magic

表の行と列は 0 から数え, 合計の行, 合計の列をそれぞれ第 M 行, 第 N 列とする. 元の表の i 行 j 列 $(0 \leq i \leq M,\ 0 \leq j \leq N)$ の値を a_{ij} とし, $L = a_{MN}$ とおく. Ka_{ij} を L で割った商を q_{ij}, 余りを r_{ij} とおくと, 変換後の表の i 行 j 列は, $r_{ij} = 0$ のときは q_{ij} で, $r_{ij} \neq 0$ のときは q_{ij} または $q_{ij} + 1$ でなければならない.

この問題は, 以下のグラフにおける最小費用流を求めることによって解ける.

s, t, u_i $(0 \leq i \leq M-1)$, v_j $(0 \leq j \leq N-1)$ を頂点とし, 以下の有向辺がある.

- (s, u_i), 容量 q_{iN}, コスト $-\infty$
- (s, u_i), 容量 1, コスト $(1 - r_{iN}/L)^2 - (r_{iN}/L)^2$ ($r_{iN} \neq 0$ のときのみ)
- (u_i, v_j), 容量 q_{ij}, コスト $-\infty$
- (u_i, v_j), 容量 1, コスト $(1 - r_{ij}/L)^2 - (r_{ij}/L)^2$ ($r_{ij} \neq 0$ のときのみ)
- (v_j, t), 容量 q_{Mj}, コスト $-\infty$
- (v_j, t), 容量 1, コスト $(1 - r_{Mj}/L)^2 - (r_{Mj}/L)^2$ ($r_{Mj} \neq 0$ のときのみ)

始点 s, 終点 t, 流量 K の最小費用流を求める.

容量 q_{ij} の各辺は, 容量 1 の辺より優先して使われるようにするためにコストが $-\infty$ となっている. また, 容量 1 の各辺は, Ka_{ij}/L を q_{ij} に丸めるか $q_{ij} + 1$ に丸めるかを表す (フローが流れる場合 $q_{ij} + 1$ に丸める).

この解法に対して, 次のような改善点が考えられる.

- s から t へのコスト $-\infty$ の辺のみからなるパスがあるので, これを予め減らす. 他の辺の容量と全体の流量を適切に減らし, u_i から v_j への容量 q_{ij}, コスト $-\infty$ の辺を削除することができる.
- 容量 q_{ij} の辺のコスト $(-\infty)$ は, 並行する容量 1 の辺のコストより真に小さければ十分である.
- すべての辺のコストを定数倍してもよい. さらに, s から t へのパスの長さがすべて 3 で一定であることから, すべての辺のコストに定数を加えてもよい. すると, すべての辺のコストを非負整数にできる.

これらを適用すると, 以下の問題に変換される.

s, t, u_i $(0 \leq i \leq M-1)$, v_j $(0 \leq j \leq N-1)$ を頂点とし, 以下の有向辺がある.

- (s, u_i), 容量 $q_{iN} - \sum_{j=0}^{N-1} q_{ij}$, コスト 0
- (s, u_i), 容量 1, コスト $L - r_{iN}$ ($r_{iN} \neq 0$ のときのみ)
- (u_i, v_j), 容量 1, コスト $L - r_{ij}$ ($r_{ij} \neq 0$ のときのみ)
- (v_j, t), 容量 $q_{Mj} - \sum_{i=0}^{M-1} q_{ij}$, コスト 0
- (v_j, t), 容量 1, コスト $L - r_{Mj}$ ($r_{Mj} \neq 0$ のときのみ)

始点 s, 終点 t, 流量 $K - \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} q_{ij}$ の最小費用流を求める.

すると, グラフは頂点数 $O(M + N)$, 辺数 $O(MN)$, 辺のコストがすべて非負となり, また, 流量は $K - \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} q_{ij} = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (Ka_{ij}/L - q_{ij}) \leq MN$ であるから, 最短路反復法による時間計算量

を $O(MN(M+N)^2)$ あるいは $O(M^2N^2 \log MN)$ と評価することができる.

ところで, この問題に対しては必ず解が存在することを証明できる. 最初のグラフにおいて, s から t への最大流が K 以上であることを示そう. それには, 最小の s-t カットの容量が K 以上であること, すなわち, 任意の s-t カットの容量が K 以上であることを示せばよい.

s-t カットに対して, s 側, t 側の頂点集合をそれぞれ S, T とおくと, カットの容量は,

$$
\begin{aligned}
&\sum_{u_i \in T} q_{iN} + \sum_{u_i \in T,\, r_{iN} \neq 0} 1 + \sum_{u_i \in S,\, v_j \in T} q_{ij} + \sum_{u_i \in S,\, v_j \in T,\, r_{ij} \neq 0} 1 + \sum_{v_j \in S} q_{Mj} + \sum_{v_j \in T,\, r_{Mj} \neq 0} 1 \\
&= \sum_{u_i \in T} \left\lceil \frac{Ka_{iN}}{L} \right\rceil + \sum_{u_i \in S,\, v_j \in T} \left\lceil \frac{Ka_{ij}}{L} \right\rceil + \sum_{v_j \in S} \left\lceil \frac{Ka_{Mj}}{L} \right\rceil \\
&\geq \frac{K}{L} \left(\sum_{u_i \in T} a_{iN} + \sum_{u_i \in S,\, v_j \in T} a_{ij} + \sum_{v_j \in S} a_{Mj} \right) \\
&= \frac{K}{L} \left(\left(\sum_{u_i \in T,\, v_j \in S} a_{ij} + \sum_{u_i \in T,\, v_j \in T} a_{ij} \right) + \sum_{u_i \in S,\, v_j \in T} a_{ij} + \left(\sum_{u_i \in S,\, v_j \in S} a_{ij} + \sum_{u_i \in T,\, v_j \in S} a_{ij} \right) \right) \\
&\geq \frac{K}{L} \cdot L \\
&= K
\end{aligned}
$$

であるから示された.

Problem G: Strange Sequence

整数列 $(x_1, \ldots, x_N)$ に対して, $x_i = y_i - y_{i-1}$ $(1 \le i \le N)$ を満たす整数列 $(y_0, \ldots, y_N)$ を考える．このような (y_i) は各項に定数を加えることで得られる不定さがあるが, $\min_{0 \le i \le N} y_i = 0$ となるものを考えることで, (x_i) と (y_i) を 1 対 1 に対応させることができる.

(x_i) が問題の条件を満たすことは, (y_i) が以下の条件を満たすことと同値である.

(1) $y_i < y_{i+A}$ $(0 \le i \le N - A)$.

(2) $y_i > y_{i+B}$ $(0 \le i \le N - B)$, すなわち, $y_i < y_{i-B}$ $(B \le i \le N)$.

(3) y_i は互いに異なる.

(4) $\min_{0 \le i \le N} y_i = 0$.

(5) $0 \le y_i \le M$ $(0 \le i \le N)$.

条件 (3), (4), (5) により, (y_i) の順序関係が定まっているならば, 1 以上 M 以下の異なる整数を N 個選ぶことで (y_i) は定まるので, (y_i) の順序関係が何通りあるかを求めて $\dfrac{M!}{N!(M-N)!}$ 倍すればよい.

条件 (1), (2) は厳しく, N が大きいとこれらを満たす (y_i) の順序関係は存在しない．例えば $A = 4$, $B = 6$, $N \ge 8$ ならば $y_0 < y_4 < y_8 < y_2 < y_6 < y_0$ により矛盾する.

$d = \gcd(A, B) (= \gcd(A, A + B))$ とおくと, 解が存在することと $N < A + B - d$ が同値であることを示す.

整数 k に対して, k を $A + B$ で割った余り (0 以上 $A + B$ 未満) を $\overline{k}$ と書くことにすると, $N < A + B$ のとき, 条件 (1), (2) は「$0 \le \overline{i} \le N$, $0 \le \overline{i + A} \le N$ を満たす任意の整数 i に対して $y_{\overline{i}} < y_{\overline{i+A}}$ であること」と同値である．$N \ge A + B$ のとき, これは (1), (2) を満たすための必要条件となる.

$e = (A + B)/d$ とおく．$\overline{0}, \overline{A}, \overline{2A}, \ldots, \overline{(e-1)A}$ には 0 以上 $A + B$ 未満の d の倍数がちょうど 1 回ずつ現れるので, 考えられる $A + B$ 個の不等式は以下のように d 個のサイクルとして書くことができる.

$$
\begin{gathered}
y_{\overline{0}} < y_{\overline{0+A}} < y_{\overline{0+2A}} < \cdots < y_{\overline{0+(e-1)A}} < y_{\overline{0+eA}} = y_{\overline{0}} \\
y_{\overline{1}} < y_{\overline{1+A}} < y_{\overline{1+2A}} < \cdots < y_{\overline{1+(e-1)A}} < y_{\overline{1+eA}} = y_{\overline{1}} \\
\vdots \\
y_{\overline{d-1}} < y_{\overline{d-1+A}} < y_{\overline{d-1+2A}} < \cdots < y_{\overline{d-1+(e-1)A}} < y_{\overline{d-1+eA}} = y_{\overline{d-1}}
\end{gathered}
$$

これらのサイクルに含まれる最大の添え字はそれぞれ $A + B - d$, $A + B - d + 1$, ..., $A + B - 1$ であるから, $N \ge A + B - d$ のときは不等号のサイクルが生じるため解が存在しない．一方, $N < A + B - d$ のときはすべてのサイクルが分断され, いくつかのチェーンとして表される．特に, この場合は解が存在する.

さて, 残る問題は $N < A + B - d$ のときに条件 (1), (2) を満たす (y_i) の順序関係が何通りあるか調べることである．条件が不等号のチェーンとして書けるという考察により, 「(y_i) のうち小さい方から j 番目はどのチェーンに属するか」を $1 \le j \le N + 1$ について定めるたびに順序関係が 1 つ定まることがわかる．よって, 各チェーンに属する y_i の個数を c_1, c_2, ..., c_L と書くと, $\dfrac{(N+1)!}{c_1! c_2! \cdots c_L!}$ 通りである.

Problem H: Read Me

いろいろな解法が考えられるが, どのデータも少なくとも 1 つの問題の入力データとしてはあり得るという条件を利用するのが簡単である.

まず, 最後の行によって B, C, E, F を識別することができる. D は 2 行目に含まれる整数の個数によって A, G から区別できる. A と G についてはフォーマットが同じなので, 値の範囲をチェックする. G の入力はすべて A の入力としてもあり得るので, G の制約を満たすかどうかのみを確認すればよい. A と G では最大テストケース数が異なるので, テストケース数も判定条件に含めることに注意を要する.