いちご 2
(STRAWBERRY 2)

解説: 平木 康傑

問題概要

- H × W のマス目に N 個のいちごがある
- 3 × 3 の範囲をとる
- とれるいちごは最大でいくつ？

問題概要

- H × W のマス目に N 個のいちごがある
- 3 × 3 の範囲をとる
- とれるいちごは最大でいくつ？

- 小課題 1 (14 点): H, W <= 1000, N <= 20
- 小課題 2 (16 点): H, W <= 1000
- 小課題 3 (29 点): H <= 10^6, N <= 500
- 小課題 4 (15 点): H = 3
- 小課題 5 (20 点): H <= 10^6
- 小課題 6 (6 点): H も W も N も大きい

小課題 1 (14 点): H, W <= 1000, N <= 20
小課題 2 (16 点): H, W <= 1000
小課題 3 (29 点): H <= 10^6, N <= 500
小課題 4 (15 点): H = 3
小課題 5 (20 点): H <= 10^6
小課題 6 (6 点): H も W も N も大きい

小課題 1 (14 点)

- H, W <= 1000 なので，とりえる 3 × 3 範囲を全部列挙できる
- 各範囲内のいちごの個数を数え，その最大値を取ればよい

- N <= 20 なので，各範囲に対してそれぞれのいちごが範囲内かを調べても十分高速
- 時間計算量は Θ(HWN) となり，間に合う

小課題 1 (14 点): H, W <= 1000, N <= 20
小課題 2 (16 点): H, W <= 1000
小課題 3 (29 点): H <= 10^6, N <= 500
小課題 4 (15 点): H = 3
小課題 5 (20 点): H <= 10^6
小課題 6 (6 点): H も W も N も大きい

小課題 2 (16 点)

- H, W <= 1000 なので，とりえる 3 × 3 範囲を全部列挙できる
- 各範囲内のいちごの個数を数え，その最大値を取ればよい

- マス目に対応する 2 次元配列をとり，各マスにあるいちごの個数を記録する
- 各範囲について，9 マスの和をとればよい
- 時間計算量は Θ(HW) となり，間に合う

- 余談: 制約が小課題 1 の上位互換なので，こちらを解くだけで小課題 1 も解けたことになり，計 30 点が得られる

小課題 3 (29 点)

小課題 1 (14 点): H, W <= 1000, N <= 20
小課題 2 (16 点): H, W <= 1000
小課題 3 (29 点): H <= 10^6, N <= 500
小課題 4 (15 点): H = 3
小課題 5 (20 点): H <= 10^6
小課題 6 (6 点): H も W も N も大きい

- とりえる 3 × 3 範囲がたくさん
  - 本当にたくさん？

- 各いちごについて，それを含む 3 × 3 範囲はちょうど 9 通り
- ということは，いちごを 1 個でも含むような範囲は高々 9N 通り
- なので，いちごを 1 個でも含むような範囲のみを列挙すれば大幅に探索範囲を絞れる

小課題 3 (29 点)

小課題 1 (14 点): H, W <= 1000, N <= 20
小課題 2 (16 点): H, W <= 1000
小課題 3 (29 点): H <= 10^6, N <= 500
小課題 4 (15 点): H = 3
小課題 5 (20 点): H <= 10^6
小課題 6 (6 点): H も W も N も大きい

- とりえる 3 × 3 範囲がたくさん
  - 本当にたくさん？
- 各いちごについて，それを含む 3 × 3 範囲はちょうど 9 通り
- ということは，いちごを 1 個でも含むような範囲は高々 9N 通り
- なので，いちごを 1 個でも含むような範囲のみを列挙すれば大幅に探索範囲を絞れる
- 実際には，被ってもいいのでとにかく各いちごについて周囲 9 範囲を調べればよい
  - 最大値が分かればいいので，同じ範囲を何回調べても支障はない
- これと小課題 1 の解法 (範囲それぞれについて，各いちごが範囲内かを見る) を組み合わせて，Θ(N^2)
- 下位互換として取れる小課題 1 と合わせて 43 点が得られる

小課題 1 (14 点): H, W <= 1000, N <= 20
小課題 2 (16 点): H, W <= 1000
小課題 3 (29 点): H <= 10^6, N <= 500
小課題 4 (15 点): H = 3
小課題 5 (20 点): H <= 10^6
小課題 6 (6 点): H も W も N も大きい

小課題 4 (15 点)

- H = 3 なので，実質 1 次元の問題
  - その代わり，横幅は広い

- 調べる範囲を 9N 通りに絞るまでは小課題 3 と同様
- 「あるマスにいちごがいくつあるか」(ある数がいくつあるか)を調べるには，次のような方針がある
  - 配列に格納してソートし，二分探索や尺取り法などを適用
  - 連想配列 (C++ なら std::map) に入れて，通常の配列と同様に座標にアクセス

- 二分探索や連想配列なら Θ(N log N)，尺取り法なら Θ(N) で解ける
- 時間制限がやや厳しいので std::vector ではなく生の配列を使う方が通りやすい
- ともかく 15 点

小課題 5, 6 (満点)

小課題 1 (14 点): H, W <= 1000, N <= 20
小課題 2 (16 点): H, W <= 1000
小課題 3 (29 点): H <= 10^6, N <= 500
小課題 4 (15 点): H = 3
小課題 5 (20 点): H <= 10^6
小課題 6 (6 点): H も W も N も大きい

- 小課題 4 の解法を 2 次元に拡張する
- 二分探索や連想配列といった方針の場合，書き換えるのは簡単
  - 2 次元座標をそのまま添え字とした配列/std::vector/連想配列を使えばそのまま拡張できる
- 尺取り法の場合は少しややこしいが，特別新しい発想は必要ない
  - 範囲をスライスして 3 分割するとラク
- 上は Θ(N log N) のまま，下は Θ(N) のままで，十分高速に動く
- 満点
- おめでとう