

## C – Just Write Numbers!

※この問題はコンテスト時間内で最適解を発見することは想定されていません。  
そのため、実行時間制限(3秒)以内でできるだけ高い得点を取れる考え方をいくつか紹介します。

No.	解答方針	目安得点
方針①	入力値をそのまま出力する	約85万点
方針②	左上から順に値を確定させる	約128万点
方針③	実行時間制限の限り以下の更新を繰り返す <ul style="list-style-type: none"><li>• 解答を一部変更</li><li>• 得点の変化を確認</li><li>• 得点が上がったら採用</li></ul>	?? 点

# 方針① 入力値をそのまま出力する

今回の問題では、出力する数値が、全て下限値 $l_{i,j}$ から上限値 $r_{i,j}$ の範囲内であれば、0より大きい得点が取れます。

この条件を満たす解答を作る簡単な方法は、入力として渡される下限値 $l_{i,j}$ 、または上限値 $r_{i,j}$ をそのまま出力することです。

今回は提出回数によるペナルティが無いため、2パターンどちらも提出すれば、得点が高いほうが順位に反映されます。

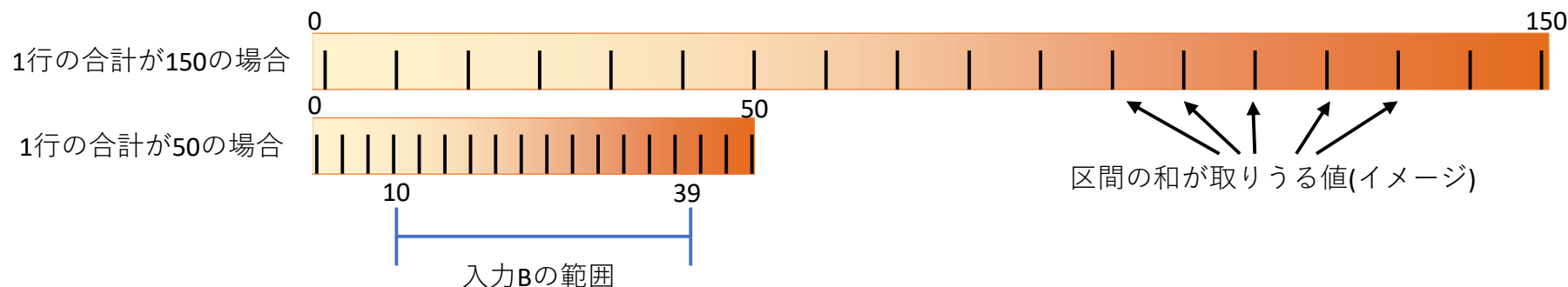
## $[r_{i,j}$ と $l_{i,j}$ でどちらが高得点か]

全て $l_{i,j}$ を出力した場合と、全て $r_{i,j}$ を出力した場合では、 $l_{i,j}$ のほうが高得点になります。これは以下のように、数値が小さいほうが高得点となる確率が高いからとなります。

$r_{i,j}$ と $l_{i,j}$ は1~9でランダムに選ばれるので、 $r_{i,j}$ から $l_{i,j}$ の間で値をランダムに選ぶと平均5となり、1行(または1列)の合計の平均は $5 \times 30$ マスで150となります。

すると、連続した数の区間の和は0~150の間に分散して存在します。

入力 $B$ の範囲は10~39で、150よりかなり小さいので、1行の合計の平均が小さいほうが、入力 $B$ の値と一致する可能性が高いことがわかります。

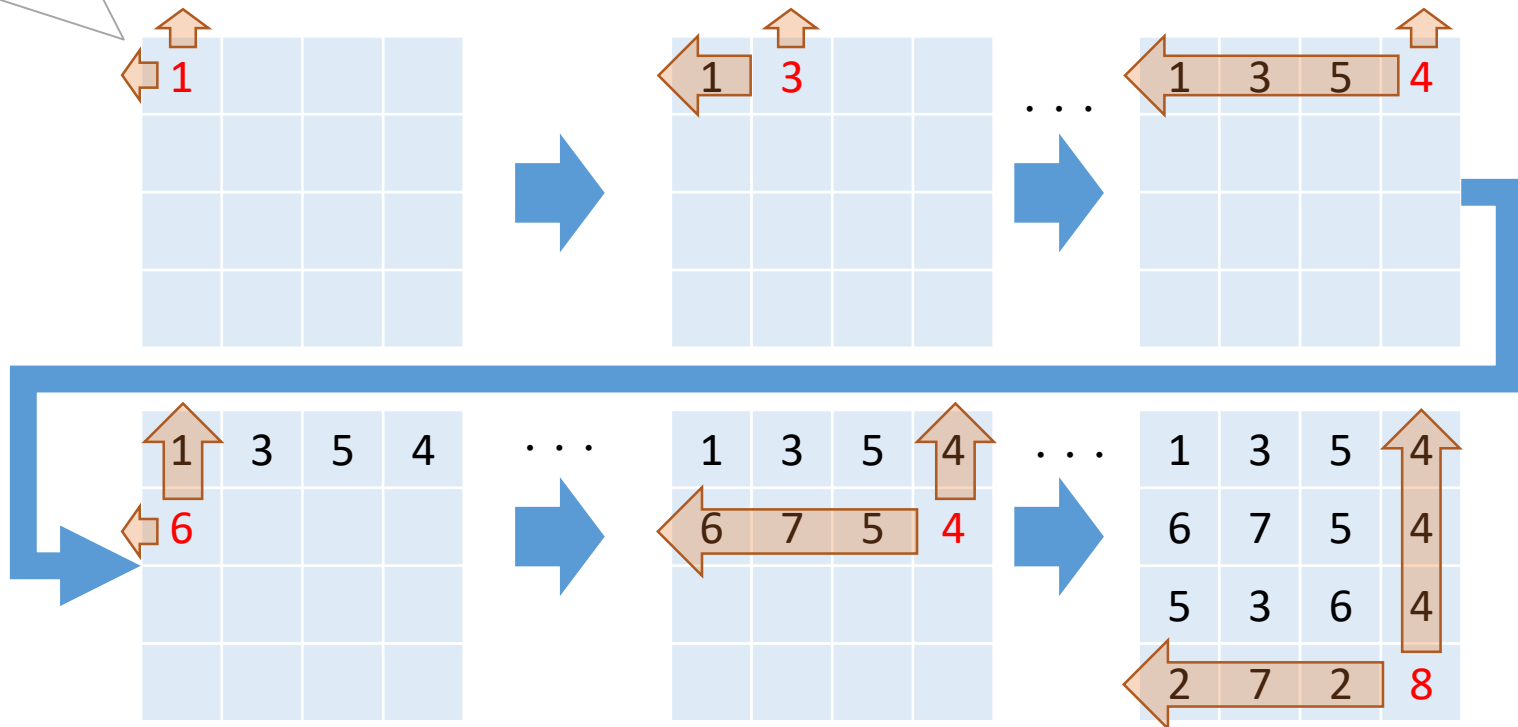


## 方針② 左上から順に値を確定させる

各マスにおいて、全てのマスとの関係を考慮して最適値を求めるのは困難です。  
そこで、左上から順にできるだけ得点の高くなる値を確定させていく方法を取ってみます。

左上から順に値を決める場合、そのマスによる得点は、  
そのマスから上に伸びる連続した数の区間と、左に伸びる連続した数の区間によります。  
それらの区間のうち、和が入力  $B$  となるものが多いほど得点が高くなります。  
これを、 $r_{i,j}$  から  $l_{i,j}$  の値で全て確認し、一番高得点になる値で確定させます。

どの値を入れても得点と同じ場合は  
また別の基準で値を決める  
(方針①を考慮し最小の値にするなど)



## 方針③ 実行時間制限の限り更新を繰り返す

まず、何らかの方法で一つの解答案を作成します。  
(方針①や②の解答から始めることになると思います)

そこから、以下の操作を、実行時間制限(3秒)の許す限り繰り返します。

1. **解答の値を一部変更**
2. 得点の変化を確認
3. **得点が高くなった場合**は変更を採用

値の変更の仕方や、採用する条件など、詳細部分の考え方により結果が変わってきます。

上記アルゴリズムは以下のような名前がついていますので、気になる方は調べてみてください。

- 山登り法(hill climbing)
- 焼きなまし法(Simulated Annealing)