ICPC国内予選 2018

17 - baton

はじめに

- 国内予選お疲れ様でした
- A-Eの解説をしますが、BとE以外は実装してないです（たぶん大丈夫）
- どうでもいいけど、先週末のコンテスト多くない？

  ukucon, SoundHound, TCOR3A, HFC, JOI Open …

まとめ

	ACチーム数	割合	解法
A	387/400	96.8%	ループ
B	140/400	35%	2次元配列操作
C	186/400	46.5%	算数
D	70/400	17.5%	枝刈り全探索
E	20/400	5%	実験
F	7/400	1.8%	幾何(軽)
G	3/400	0.8%	—
H	1/400	0.3%	木DP？

A問題

所得格差

A - 問題概要

- 長さNの数列{A_i}が与えられる
- 平均値**以下**の要素数を求めよ

A - 解法

- 総和を求めて、Nで割れば平均値が求まるので、
  これと全てのA_iを比較して条件を満たす要素数を数える
- ”以下”なので、平均値は切り捨てで良い(整数演算でOK)
- 今回は必要ないが、誤差を避けるためには、

$$A_i \cdot N \leq \sum_{j=1}^{N} A_j$$

のように、整数だけで比較すると良い

B問題

折り紙

B - 問題概要

- N × Mの長方形の紙で折り紙をする
- 「上からy cm」or「左からx cm」の線で谷折りを繰り返す

※問題文は左下が(0,0)だけど、左上としても一緒

- 全ての指示に従って折り終わったあとに、左上から(x+1/2,y+1/2)の位置に重なっている紙の枚数を求める

B - 解法

1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0

B - 解法

1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0

B - 解法

0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0

B - 解法

0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0

B - 解法

0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	4	4	4	2	2	2	0	0	0	0
0	0	0	4	4	4	2	2	2	0	0	0	0
0	0	0	4	4	4	2	2	2	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0

B - 解法

0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	4	4	4	2	2	2	0	0	0	0
0	0	0	4	4	4	2	2	2	0	0	0	0
0	0	0	4	4	4	2	2	2	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0
0	0	0	2	2	2	1	1	1	0	0	0	0

B - 解法

0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	4	6	4	4	0	0
0	0	0	0	0	0	0	4	6	4	4	0	0
0	0	0	0	0	0	0	4	6	4	4	0	0
0	0	0	0	0	0	0	2	3	2	2	0	0
0	0	0	0	0	0	0	2	3	2	2	0	0

B - 必要なこと

- 各作業前(折る/重なりを求める)に左上の位置を求める

  → sum[i][j] > 0を満たす辞書順最小の(i,j)

- 折る位置を対称として配列の値を移動させる

  → sum[i][c+j] += sum[i][c-j-1]; sum[i][c-j-1] = 0;

B - 実装の注意

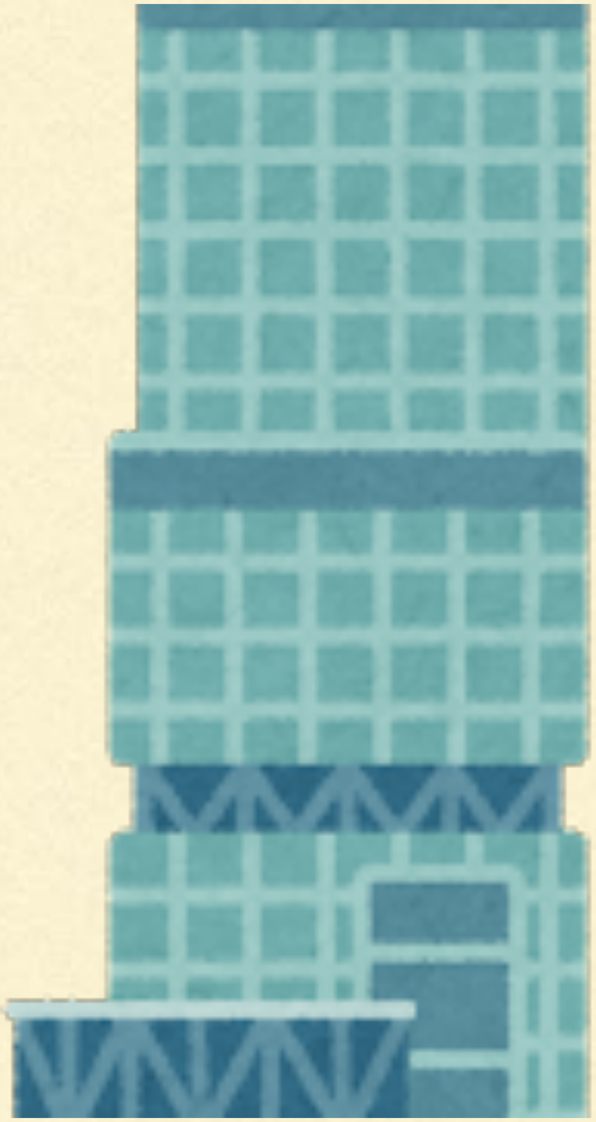
- 紙の左上を配列の[0][0]に毎回合わせるのは面倒
  そのため、左上の位置を毎回求めるほうが楽
- ただ、はみ出るような折り方を繰り返すと多くの配列が必要
  →600 x 600 くらいの配列を確保すれば十分
- 最大でも32x32なので1024よりも大きい結果が出力されたらおかしい(提出前に確認してみると良い)

C問題

超高層ビル

「みなとハルカス」

あべのハルカス→



C - 問題概要

- 連続する整数の和が X になるような 区間を考える

  例) 1+2+3+4 = 10、 6+7+8 = 21、 3 = 3 …

- そのような区間の中で最長の長さの区間を求めよ

C - 解法

- 一つの式で答えを出すのは難しそう
- 区間の最小値を固定すると最大値も求まるけど候補が10^9
- なら、長さを固定してみると良さそう

C - 解法

- 長さを固定すると…

  候補はO(√N) (なぜなら、1+2+…+A = A*(A+1)/2)

- さらに、長さを固定すると簡単に区間を求められることが分かる

C - 解法

- 長さLが奇数の時 -> 中央値 * 長さ = 総和

  そのため、X % L == 0 のとき、解が存在

- 長さLが偶数の時 -> 中央値 * 2 * L/2 = 総和

  そのため、X % (L/2) == 0のとき、解が存在

- ただし、両方とも最小値が1以上であることをチェック

C - 解法

- 長さごとにチェックができるようになったので、
  解が存在する最大のLが答え

- Lが決まれば区間内の最小値も一意に決まるので、
  それも一緒に出力する

D問題

全チームによる
プレーオフ

D - 問題概要

- 総当たりリーグ戦で全チームを同順位にしたい
- すでに行われた試合の結果が与えられるので、全チームが同順位になるための、残りの試合結果の組み合わせ数を求める
- ただし、各試合では引き分けはなく、
  リーグ戦の順位は各チームの勝ち数のみで決まる

D - 解法

- とりあえず入出力例をみてみよう
- 1試合しか確定していないので、明らかにこれは最大
  →全探索できそう

INPUT	OUTPUT
9 1 1 2	1615040

D - 解法

- 本当に答えの最大値が1615040だと、全探索できるのか？
- 試合の組み合わせは、最大で2^35なので枝刈りは必要
- 再帰関数で探索していくときに、探索先に解が1つ以上必ず存在するように潜っていくと、O(答え*深さの最大)になる
→今回は必ず解が存在する条件チェックは難しそう？

D - 解法

- それでも、各チームの勝ち数が(N-1)/2になるようにチェックしていけば十分高速

- なぜ？

  ->空のリーグ表で考えて見よう

D - 解法

	A	B	C	D	E	F	G	H	I
A									
B									
C									
D									
E									
F									
G									
H									
I									

半分は勝たないと行けないので、組み合わせは、${}_8C_4 = 70$

D - 解法

	A	B	C	D	E	F	G	H	I
A									
B									
C									
D									
E									
F									
G									
H									
I									

1試合の結果は決まっているので、最大でも $_7C_3 = 35$

D - 解法

	A	B	C	D	E	F	G	H	I
A									
B									
C									
D									
E									
F									
G									
H									
I									

2試合の結果は決まっているので、最大でも ${}_6C_3 = 20$

D - 解法

- つまり、最後の行までこれを繰り返すと、

$$ {}_8C_4 \cdot {}_7C_3 \cdot {}_6C_3 \cdot {}_5C_2 \cdot {}_4C_2 \cdot {}_3C_1 \cdot {}_2C_1 \cdot {}_1C_1 $$

$$ = 70 \cdot 35 \cdot 20 \cdot 10 \cdot 6 \cdot 3 \cdot 2 \cdot 1 = 17640000 $$

- 間に合う！！

D - 実装

- 常に、全てのチームの勝ち数が、半分以下であるように順番に勝ち負けを決めていけば十分
- dfs(注目している試合, 各チームの勝ち数)

  -> dfs(勝ちの場合) + dfs(負けの場合)
- 各チームの勝数は変化したところ(つまり2チーム)だけ注目すれば十分(毎回全チェックしても国内予選では問題無い)

E問題

浮動小数点数

E - 問題概要

- 仮数部52bit 指数部12bitの浮動小数点数が与えられる
- この浮動小数点数を桁落ち誤差を考えながら、
  0にN+1回足したときの値を求めよ

E - 問題整理

- m を残る 52 ビットが表す二進小数に 1 を加えた $(1.b_{52}...b_1)_2$ とする
- →値を扱いやすいように、常に53bit目に1を入れて扱う

E - 問題整理

- 足し算はどうやって行われているのか？

$$
\begin{array}{rr}
10000000000...0000000000 & (e = 0) \\
+ & \\
10000000000...0000000000 & (e = 0) \\
\| & \\
100000000000...0000000000 & (e = 0) \\
\| & \\
10000000000...0000000000 & (e = 1)
\end{array}
$$

E - 問題整理

- 足し算はどうやって行われているのか？

$$
\begin{array}{rl}
10000000000...0000000000 & (e = 2) \\
+ & \\
10000000000...0000000111 & (e = 0) \\
\| & \\
10000000000...0000000000 & (e = 2) \\
+ & \\
100000000...0000000001 & (e = 2) \\
\| & \\
10100000000...0000000001 & (e = 2)
\end{array}
$$

E - 解法

1. s := a
2. for n times {
3.   s := s + a
4. }

- sの指数部の値に合わせてaをビットシフトして足しているのと同じ
- 繰り上げが起こるまでは、同じ値を足し続けるので、繰り上げが起こるまで一気に足しても問題無さそう
- 53回右シフトすると0になるので、足す回数も最大で53回

E - 実装

- a := 入力の53bit目を1にした値(long longに収まる)
- 最初に、s = a、e = 0
- 繰り上がりするまでに足せる回数は、

  (2^53 - s - 1) / a + 1 で求まる
- 繰り上がったら、e += 1、a /= 2

宣伝

- 2018/7/21 01:00 - 7/24 01:00 に ICFPCが行われます
- マラソン形式です

  一昨年 -> 折り紙、去年 -> ゲーム(AI作成)
- チームで参加できるので参加しませんか？(募集中)
- https://icfpcontest2018.github.io