第30回世界コンピュータ将棋選手権

きふわらべ アピール文書

2020年03月30日 高橋智史

CODING

コーディング・テクニックをPR**するのね**

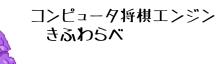
DARE 誰なんだぜ?



開発者 高橋 智史

> 「わたしは よく ミスのあるコーディング をしてしまう☆ そこで、 ミスが少なくなるコンピューター将棋のコーディング について書きたいと思う☆」





「 今年も わたしをPR する内容が無いのか……☆」



「 ミスが少なくなってから 書きましょうよ! これから ミス するのに……!」

岡崎夢美 /TOHO PROJECT FANMADE ※

※北白河ちゆり、岡崎夢美は 東方夢時空 の登場キャラクター/ ⑥上海アリス幻樂団 様の著作物です。

符号は そのまま使おうぜ☆

DAI ICHI SYOGEN

180°回転した 第1象限 ね。



プラス マイナス

ι +-を間違えんなよ、という意味で しょうげん

象限 という言葉がある☆

だいいちしょうげん

チェスなら 第1象限、

じょうげはんてんした だいいちしょうげん

国際囲碁や連珠なら 上下反転した第1象限、 ひゃくはちじゅうどかいてんした だいいちしょうげん

将棋なら 180°回転した第Ⅰ象限 だぜ☆」



みな バラバラ で わらう☆」



第 I 象限しか 使わないでしょ。 負数 を使いたくなるときなんて あるの?」

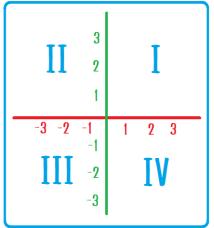


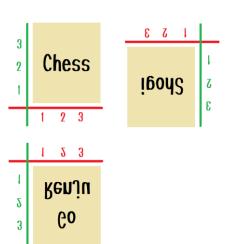
「 象限の考え方は あとで使う☆ 今は 将棋盤は 下手(先手)視点で 印刷するのが慣習なんで 180°回転した第Ⅰ象限に 従おうぜ☆」

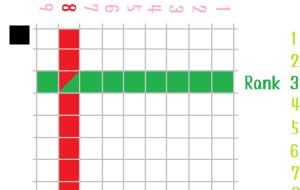
2

6 7 8

Orthant







File(列) Rank(行)

タテ棒の ヨコ位置が ファイル(<mark>筋</mark>) で、 ヨコ棒の タテ位置が ランク (段) なのよね」



「 そうそう……、 ファイル、ランク の順に並べて 足し算でも 掛け算でもなく ただ並べる "辞書順" という 操作をして アドレス(番地)を マッピングしようぜ☆」

File Rank

8 + 3 = 11

8 3 = 83

Mapping(地図作り)



「直交で配置すると 右図の通りだぜ☆」



「数が 縦書きの順に並んでるな☆」



「番地って、符号と おんなじね」



「右図を見ると

北隣 は - 1 、 東隣 は -10 、 南隣 は + 1 、 西隣 は +10



「 今年は何が 世界一周 するんだぜ☆?」



「盤の外に 飛び出たかどうかの 判定が 重要なのよ」

	1	Z	3	4	5	6	'/	8	9
11	11	21	31	41	51	61	71	81	91
2	12	22	32	42	5 2	6 2	72	82	92
3	13	23	33	43	53	6 3	73	83	93
4	14	24	34	44	54	64	74	84	94
5	15	25	35	45	55	65	75	85	95
6	16	26	36	46	56	66	76	86	96
7	17	27	37	47	57	67	77	87	97
8	18	28	38	48	58	68	78	88	98
9	19	29	39	49	59	69	79	89	99

North

				•						
	9	8	7	6	5	4	3	2	1	
	91	81	71	61	51	41	31	21	11	1
	92	8 2	72	6 2	5 2	42	3 2	22	12	2
	93	83	73	6 3	53	43	33	23	13	3
E.	94	84	74	64	54	44	34	24	14	4
est	95	85	75	65	55	45	35	25	15	5 08
+	96	86	76	66	56	46	36	26	16	6 +
	97	87	少	67	5 7	47	37	27	17	7
	98	88	78	68	58	48	38	28	18	8
	99	89	79	69	59	49	39	29	19	9

South

10 9 8 7 6 5 4 3 2 1 0

0	00	10	20	30	40	50	60	70	80	90	100
1	01	11	21	31	41	51	61	71	81	91	101
2	02	12	22	3 2	42	52	6 2	72	82	92	102
3	03	13	23	33	43	53	6 3	73	83	93	103
4	04	14	24	34	44	54	64	74	84	94	104
5	05	15	25	35	45	55	65	75	85	95	105
6	06	16	26	36	46	56	66	76	86	96	106
7	07	17	27	37	47	57	67	77	87	97	107
8	08	18	28	38	48	58	68	78	88	98	108
9	09	19	29	39	49	59	69	79	89	99	109
10	10	20	30	40	50	60	70	80	90	100	110

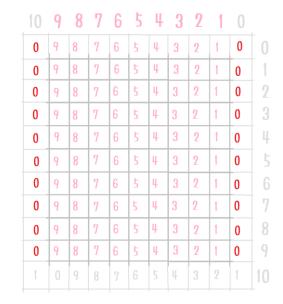
Sentinel node(番兵)



幸いなことに 将棋盤では 〇筋、〇段 は未使用なので これを使える☆」

アドレスわるじゅうきりすて をじゅうでわったあまり

adr/10 mod 10





「 真横に 1マス はみ出していれば 十の位が ○ になっているぜ☆」



| 桂馬は | 敵陣の奥から | 1段目 | 2段目に 居ることはないから | 横を調べるだけでいいのね」



「縦に 1マス はみ出していれば ーの位が ○ になっている☆」



「こんなの」計算1つで ポン と調べられるのかだぜ☆?」



「ランク10のファイル 以外は、 セルに セル自身の ファイルと ランクを お手軽に マッピングできるので……☆」

アドレス をじゅうでわったあまり

adr mod 10

10	9	8	7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	0	0	0	0
1	1	1	1	- 1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9	9	9	9
0	0	0	0	0	0	0	0	0	0	0	10

$(|adr| \mod 10 != 0)$ and $(|adr/10| \mod 10 != 0)$

タテへ飛び出ていないか判定

ヨコへ飛び出ていないか判定



「ポン と1つでは行かないが、全方向これでいけるのは魅力だぜ☆」



「ただし きふわらべ は 既に置いてある 2ー の桂を O九 に 飛ばそうとするので やはり チェックは入れなくてはいけない☆」



「 なんで 相手の駒を動かすの!」



「 動かせるのに……☆」

GENGO

言語を 使いましょう

anata あなた の あなた は、 わたし だぜ☆



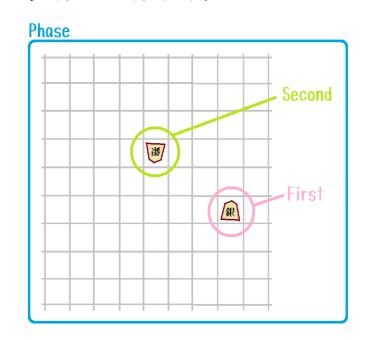
「 先手を First、 後手を Second と 呼ぶことにしようぜ☆?」



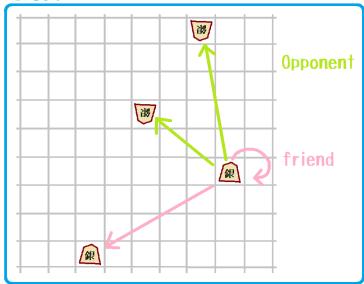
「Black と White を 使わないのか☆?」



「わたしが覚えられない☆」



Person





「自分から見て 自分の駒を Friend、 相手の駒は Opponent な☆」



Opponent の Opponent は 他人だろ☆」



2人しかいないゲームでは 自分以外は 相手ねえ」



「分かりやすくするために、手番を持ってる方を Friend とするような視点で 探索部を組めだぜ☆ つまり、

First や Second を使わずに、 Friend や Opponent を使って プログラムしろだぜ☆」

SENTE

GOTE

先手があれば 後手は要らないぜ☆

HYAKU HACHI JYU DO KAITEN

盤を 180°回転 しましょう



「 で……、後手の銀は こう 5方向に 動けるわけだが……☆」



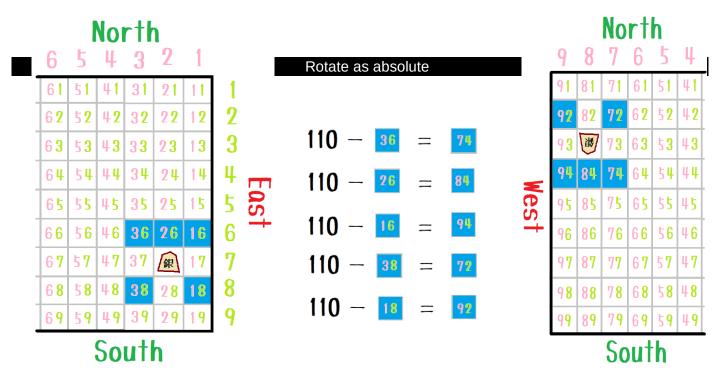
「盤の両端が 1列目、9列目で 足した数が 10列目 であることは 日ごろ10進数を使う わたしたちには 都合がいい☆」

North 5 4 3 2 1 71 61 51 41 31 21 11 91 81 72 | 62 | 52 | 42 | 32 | 22 | 12 92 82 3 73 63 53 43 33 23 13 93 83 94 84 74 64 54 44 34 24 14 95 85 75 | 65 | 55 | 45 | 35 | 25 96 86 76 66 56 46 36 26 16 97 87 77 | 67 | 57 | 47 | 37 **| 銀** | 17 8 78 68 58 48 38 28 18 98 88 99 89 79 69 59 49 39 29 19 9

South



「基数10 から ある数9 を引くと 補数1 が 出てくるんだが、 平面で 全体の数 から ある数 を引くと 180°回転した数になる☆ やってみようぜ☆?」





「先手で プログラムを組んでおいて、 後手は 110 から引けばいいのね」



「分岐とか引き算とか気になって 先手用と 後手用の2つを 組んでみたんだが、 コードが長いと それを読んでいる 自分の寿命 が 足りなくなるんで 同型は おまとめして コードを短くしようぜ☆?」

相対を使いましょう

CHUSHIN

わたしを 中心 とする世界だぜ☆



「相対番地というのは、 現地点からの +- だぜ☆」

Rotate as relative

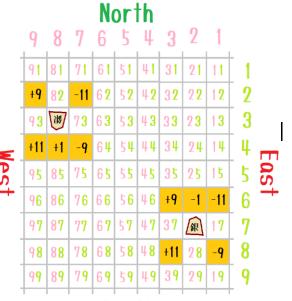


「後手は 先手の頭に マイナスを 付けるだけでいいのか☆」

+ <mark>72</mark>						+12	+2	-8	-18							-88	
	+63					+13	+3	-7	-17						-77	-87	
		+54				+14	+4	-6	-16					-66		-86	
			+45			+15	+5	-5	-15				-55			-85	
				+36		+16	+6	-4	-14			-44				-84	
					+ 27	+17	+7	-3	-13		-33					-83	
						+18	+8	-2	-12	-22						-82	
						+19	+9	-1	-11							-81	
+80	+70	+60	+50	+40	+30	+20	+10	銀	-10	-20	-30	-40	-50	-60	-70	-80	
							+11	+1	-9							-79	
									-8	-18						-78	
									-7		-27					-77	
									-6			-36				-76	
									-5				-45			-75	
									-4					-54		-74	
									-3						-63	-73	
									-2							-72	



「この式を覚えるか、理屈を理解すれば 表を丸暗記する必要は無いからな☆」



South



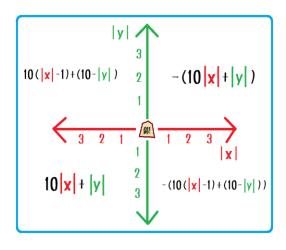
全部は埋めたくないが、 盤のどこにいても 17×17の相対番地は 使える☆」



東へ3 南へ3 と言う代わり -27 の1回で済むのね」



「 そんなケチらんでも……☆」





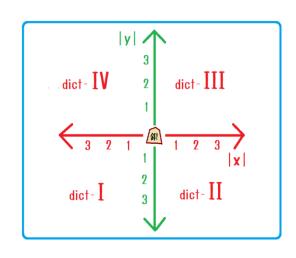
「呼び方もないと 不便なんで 名前を付けてやろうぜ☆?

じしょ しょうげん

辞書象限 なんて どうだぜ☆?」



「すぐ勝手な真似するの わらう☆」





「この相対番地の空間では、 正負の符号を反転させるだけで 180°回転できるぜ☆」



「相対番地を使うと コード量が 減んの?」



「北西 という言葉が 10(|x|-1)+(10-|y|) という式になり、 先後反転 という言葉が - になったと考えてくれだぜ☆」

\vdash								
+36		+16	+6	-4	-14			-44
	+ 27	17	+7	-3	-13		-33	
	下	+18	+8	-2	-12	-22		
		+19	+9	-1	-11			
+40	+30	+20	+10	銀	-10	-20	-30	-40
			+11	+1	-9			
					-8	-18		
					-7	1	-27	
					-6			-36

North

	9	8	7	6	5	4	3	2	1	
	91	81	71	61	51	41	31	21	11	1
	+9	82	72	6 2	5 2	42	32	22	12	2
	93	器	73	63	53	43	33	23	13	3
E	+11	+1	74	64	54	44	34	24	14	4,
PC	95	85	75	65	55	45	35	25	15	5 8
+	96	86	76	66	56	46	36	-1	-11	6
	97	87	77	6 7	5 7	47	37	銀	17	7
	98	88	78	68	58	48	38	28	-9	8
	99	89	79	69	59	49	39	29	19	9
				_					'	

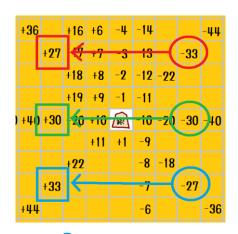
South



将棋の駒はシンメトリーなので、 東へ動かすついでに 西へ動かすことも一緒にセットにして コードを 減らしたくないかだぜ☆?」



変だぜ☆」





例えば -33 を 27 にするような 東西反転は、

-33 のマイナス符号を消して 第1辞書象限にある 33 に変身させ、 下図のアルゴリズムを使って -27を出し、 最後に、安いたマイナス符号を戻して

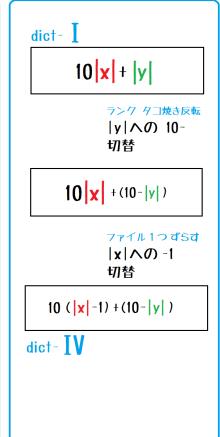
最後に 省いたマイナス符号を戻して 27 という答えになるぜ☆」

dict-I から

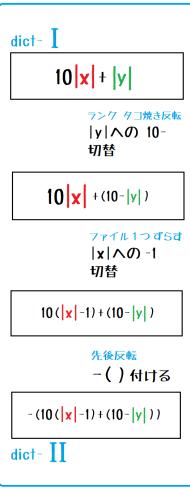
180°回転



南北反転



東西反転



※dict-I の番地にするときに負の符号を省いたら、最後に忘れずに付け直すこと。



象限は こうやって 使うのねぇ」



未認可な数学を 勝手に作ってるけどな☆」

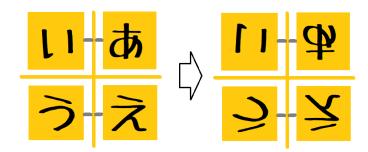


東西反転って、南北反転より 手数かかるんだな☆ 整数の順序に沿ってない方向だからかだぜ☆ おもしろ……☆」

+36	+ 2 6	+16	+6	-4	-14	-24	-34	-44
+37	+ 27	+17	+7	-3	-13	-23	-33	-43
+38	+ 28	+18	+8	-2	-12	-22	-32	-42
+39	+29	+19	+9	-1	-11	-21	-31	-41
+40	+30	+20	+10	銀	-10	-20	-30	-40
+41	+31	+21	+11	+1	-9	-19	-29	-39
+42	+32	+22	+12	+2	-8	-18	-28	-38
+43	+33	+23	+13	+3	-7	-17	-27	-37
+44	+34	+24	+14	+4	-6	-16	- 2 6	-36



「ファイルをなんで 1 ずらすかというと、 真ん中のタテ1列を 多めに持ってる方が あるから、その調整だぜ☆」





「 タコ焼き反転というのは 一の位だけを ひっくり返す ことを言っていて、 空間全体ではなく、タコ焼き……、部分の盤を ひっくり返すことだぜ☆」

FUSU

AMARI

負数 を割った 余り が どうなるかは 未定義だぜ☆

ZETTAICHI

絶対値 を割ればいいのよ



マイナスいち を じゅうでわったあまり

r −1 mod 10

は いくつだぜ☆?」



「Wolfram alpha だと 9 だぜ☆ Rust だと -1 だぜ☆」



「 結果が違う☆」

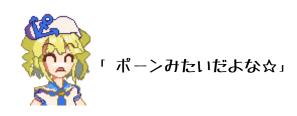


「 結果がバラけるのが嫌なら 絶対値にしてから mod しろだぜ☆」



-1 と 9 じゃ、 北と 北西ぐらい ちがうんだけど」





HYAKUHACHIJUDO KAITEN NANBOKU HANTEN

180°回転 南北反転で十分ね

TOUZAI HANTEN

180°回転して 東西反転 したいときあるけどな☆



「これで、北を180°回せば南∧☆ 真ん中では 南北反転しても同じだぜ☆」



「 北の -1 は 第111辞書象限 なの? 第IV辞書象限 なの?」



「境界は軸の片方が〇で、それが効いて どっちの象限でも 結果は同じだぜ☆」













36	26	16
37	銀	17
0.0		4.0



36	-1	16
37	銀	17
38	28	18

36	26	16
37	銀	-10
38	28	18







東を180°回せば西へ☆ 南北反転しても 変わらないぜ☆」



銀

28

36 26

37

38









なんで 南北の南北反転は機能が被るのに 東西の南北反転は機能が被らないの?」



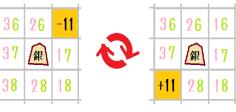
そんな、なぜ鏡は左右反転するのに 上下反転しないのか、 見たいに言われても……☆」



あれって、人間は左右対称にできていて、上下対称にはできていないことを 自己言及 してるだけだよな……☆」



「北東は 180°回せば 南西へ☆ そこから 南北反転したら 北西へ☆ そこから 180°回せば 南東へ☆ そこから 南北反転したら 北東へ☆」







ſ	90°ずつ回せばよくない?」	-		
		36	26	1
		37	銀	1
		20	0.0	





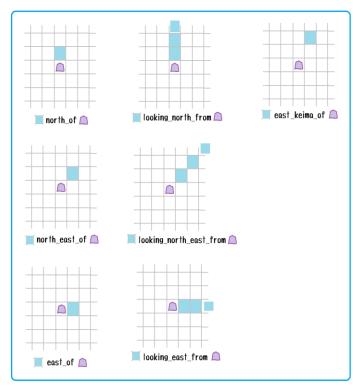
「 桂馬の動きは 90° 回転では 作れないな☆」

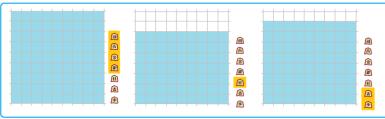


「右図の7つの動きがあれば 東西反転、先後反転で すべてカバーできるな☆ 持ち駒 打つの以外☆」



「 打つのも コーディング しましょう!」







ランク1、ランク2 と それ以外を 見分けられる いいマッピングが あるのかだぜ☆?」



「 単に ランク 見た方が 早いかも……☆」



「13 から 20 の手前まで for ループして、終わったら 出口から出てきた 20 に 3 足して また for ループ繰り返せば いいの?」



「このマルチ・コア 非同期処理 の現代、 ループ・カウンターをケチるより マッピングを練習した方が いいと思う☆」

	1	2	3	4	5	6	7	8	9
-									
_									
3	13	23	33	43	53	6 3	73	83	93
4	14	24	34	44	54	64	74	84	94
5	15	25	35	45	55	65	75	85	95
6	16	26	36	46	56	66	76	86	96
7	17	27	37	47	57	67	77	87	97
8	18	28	38	48	58	68	78	88	98
9	19	29	39	49	59	69	79	89	99



「まあ、寿命も足りなくなるしな☆」

「 アドレス (番地) さえ出てくれば、絶対的な先後反転 も使えるぜ☆」

KYUJU DO

90° は 軸を使った空間の 基本だぜ☆

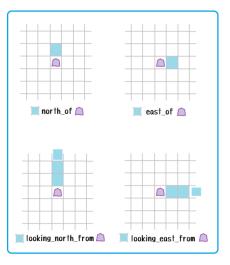
RYUJU DO HAN TOKEI MAWARI 東を 90° 反時計回り したら 北 じゃない?



「思ったんだけど、90°回転させれば 北と 西も 片方 省けない?」



「マッピングの式を思いつけば 一発だぜ☆」



-4		-14	-24	-34	-44
	40	39	38	37	36
-3		-13	-23	-33	-43
	30	29	28	27	26
-2		-12	-22	-32	-42
	20	19	1 8	17	16
-1		-11	-21	-31	-41
	70	9	∞	7	6
銀		-10	-20	-30	-40
	熟		- 2	မ	=

d5 **d**5



「あっ! なんか この式 見たことある!」



「 地道に 筆算して 合わすんだな☆」



「全部のマスで 答えが合ってる計算式を見つければ OK だぜ☆ X軸と Y軸 を入れ替えることに 気づくのは 筋がいいぜ☆」

- 0 4

40

1 0

40



図を描いてみればいいのよ。 将来 反時計回りしたいなら 時計回りの過去と 比べるのよ」



「う〜ん? 10で割ればいいのかと 思ったけど……☆」



「まず 1桁目 を 合わせてみろだぜ☆」

-4		-14	-24	-34	-44
	0 #	39	38	37	36
-3		-13	-23	-33	-43
	30	29	28	27	26
-2		-12	-22	-32	-42
	20	19	18	17	16
-1		-11	-21	-31	-41
	1 0	9	∞	7	6
銀		-10	-20	-30	-40
	鶏		- 2	-3	+

- 1 4 \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \
4
$\frac{-\begin{array}{c} 1 \\ 3 \end{array}}{3}$
1 0 - 1
9

-4	-14	-24	-34	-44	
40	39	38	37	36	
-3	-13	-23	-33	-43	
30	29	28	27	26	
-2	-12	-22	-32	-42	
20	19	18	17	16	
-1	-11	-21	-31	-41	
10	9	∞	7	6	
銀	-10	-20	-30	-40	
鶏		- 2	ယ	=	



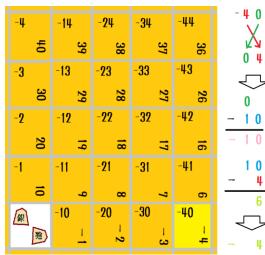
真北も ランクが繰り上がって うまく いくけど……」

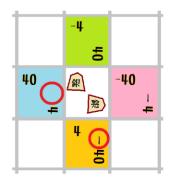


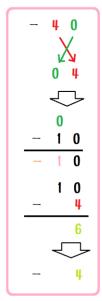
「よくできてるもんねぇ」

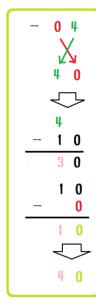


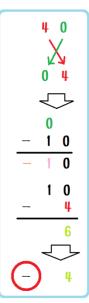
「規則性は 突然 破綻することが よくある☆ だから 一通り調べなければ いけないぜ☆」

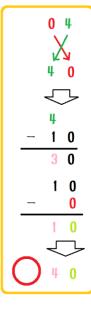














「正負の符号が 合わないケースだって あるぜ☆」



「 じゃあ 気に入らないけど、 adr が 正の数 だった場合、 計算結果を マイナス にして 符号を反転させましょう」



「式を立てたぜ☆」

自分の西側から北側へ反時計回りに9〇°回転(負のmodが符号をキープするなら):

 $((adr mod 10) - 1) \times 10 + (10 - (adr / 10 mod 10))$

ランクがファイルへ

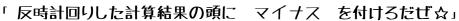
ファイルがランクへ



result if the adr is a positive number result otherwise



「ミラー(東西反転)してるときは、 時計回り にしたいんだけど」





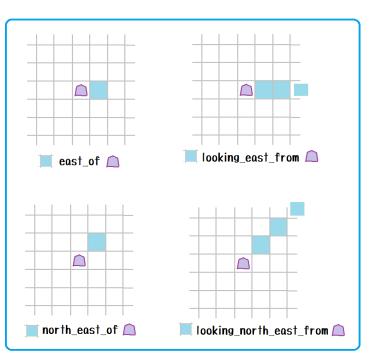
「 どんどん 難読化コード になっていってて わらう☆」

YONJUGO DO

45° だって いけんじゃないの?

MA

ま、 やってみたら いいんじゃないか☆?





「 反時計回りに 45°回転は できないの?」



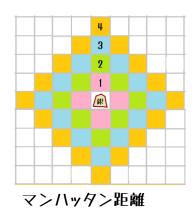
マッピングできれば できるし、 マッピングできなければ できないぜ☆」

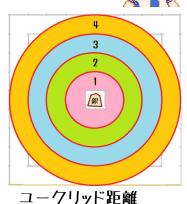


マンハッタン距離だという ことに注意しろだぜ☆」



わたしがシェビチェフ距離でも 使うと思ってるのかだぜ☆?!









これが……☆」



こうなればいいのよ☆」



「これでは 時計回りだが 描き直すのが めんどくさい☆ とりあえず このまま 一周分 絵にしてみようぜ☆?」

				-4	-14	-24	-34	-44
				-3	-13	-23	-33	-43
				-2	-12	-22	-32	-42
				-1	-11	-21	-31	-41
				銀	-10	- 20	-30	-40
-4	-14	-24	-34	-44	-43	-42	-41	-40
-4				-44 -33	, ,	, -		-40
-4		-13	-23		-32	-31		-40
-4		-13	-23 -12	-33	-32 -21	-31		-40

+36	+ 26	+16	+6	-4	-14	-24	-34	-44
+37	+ 27	+17	+7	-3	-13	-23	-33	-43
+38	+ 28	+18	+8	-2	-12	-22	-32	-42
+39	+29	+19	+9	-1	-11	-21	-31	-41
+40	+30	+20	+10	銀	-10	-20	-30	-40
+41	+31	+21	+11	+1	-9	-19	-29	-39
+42	+32	+ 22	+12	+2	-8	-18	-28	-38
+43	+33	+23	+13	+3	-7	-17	- 27	-37
+44	+34	+24	+14	+4	-6	-16	- 26	-36

-4	-14	-24	-34	-44	-43	-42	-41	-40
+6	-3	-13	-23	-33	-32	-31	-30	-39
+16	+7	-2	-12	-22	-21	-20	-29	-38
+ 2 6	+17	+8	-1	-11	-10	-19	-28	-37
+36	+ 27	+18	+9	銀	-9	-18	-27	-36
+37	+ 28	+19	+10	+11	+1	-8	-17	- 2 6
+38	+29	+20	+21	+ 22	+12	+2	-7	-16
+39	+30	+31	+32	+33	+ 23	+13	+3	-6
+40	+41	+42	+43	+44	+34	+24	+14	+4



「フーム☆ 180°回転ルールは 生きてそうだな☆ じゃあ 半分だけ 調べればいいか……☆」

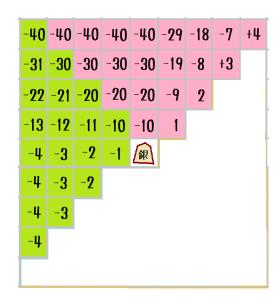


引く数

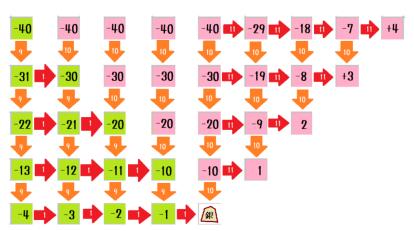
引かれる数



「上右の図から、上左の図を引いて 差を取ったのが 左図 だぜ☆ 何か規則性が ありそうだな☆」



差



-4 -3 -2

-4 -3

-4



かんかく

間隔 を見てみようぜ☆ すべての 間隔 を ○ にするルールを見つけようぜ☆ 混合しても構わない☆ 例えば……☆」





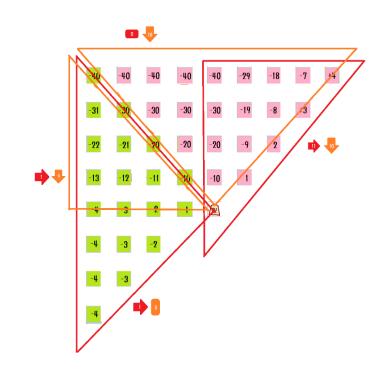
「ぱっと見た感じ 放射状に 4つのセクションに 分かれているから☆」

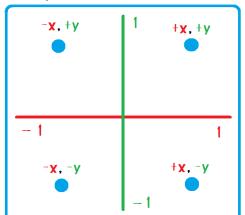


「 4つに区切って 別々の 式を当てはめて すべての間隔を O にしようぜ☆?」



「このセクションって どうやって 区別するの?」







「×、yの符号を見れば 4つには分けられるから、 あとは その中を 半分こ に するだけだぜ☆」



「 絶対値を比較するぜ☆ 境界線上が どっちのセクションか 分からんな……☆」



「だが ぼんやり と 方針は見えたので 詰めていこうぜ☆?」



「 こんなんで 何が見えるんだぜ☆! - 早押しクイズか☆!」



「呼び方もないと 不便なんで 名前を付けてやろうぜ☆?

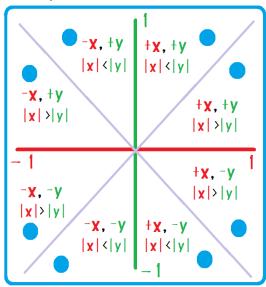
じしょ はんしょうげん



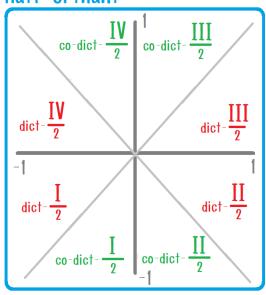


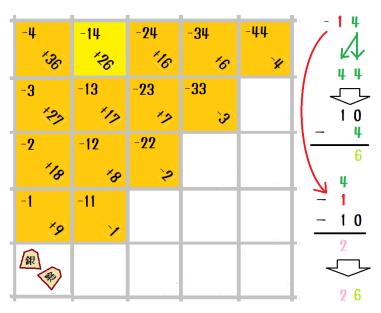
「 すぐ勝手な真似するの わらう☆」





Half orthant





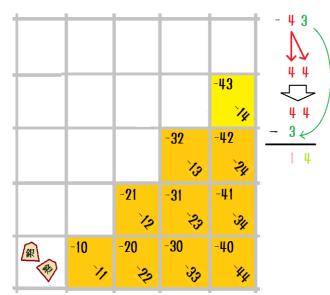


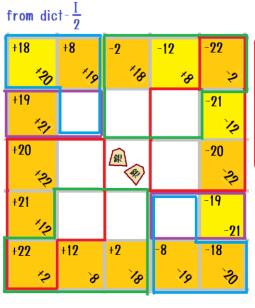


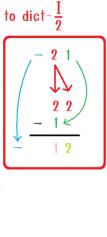
だいさん じしょ はんしょうげん dict-III/2 では シンプルだな☆」

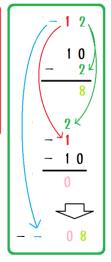


「 そんなものは 無いのに……☆」

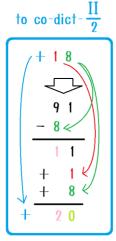


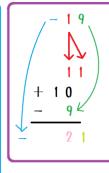






to co-dict- $\frac{I}{2}$





to dict- $\frac{II}{2}$



「ここから先 検証はまだ 十分 ではないが、 だんだんと 雰囲気が つかめてきただろ☆ 回転というのは ×軸と y軸 の入れ替え方 と 向き だぜ☆」

「こんなこと やろうとするから バグるのでは……☆?」

dict I/2;

$$10|x|+|y|$$

co-dict I/2;

$$10(|x|-|y|) + |x|$$

co-dict II/2;

$$-(10(|y|-|x|-1)+(10-|y|))$$

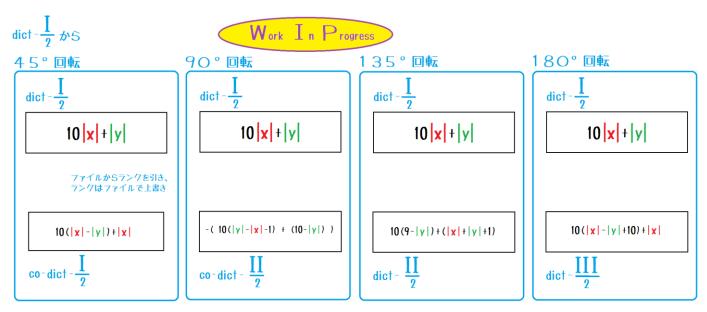
dict II/2;

$$10(9-|y|) + (|x|+|y|+1)$$

dict III/2;

$$10(|x|-|y|+10) + |x|$$

※省いた元の負の符号は 最後に付け直すこと。



※未検証 時間切り

YOHAKU

25ページでは 余白が足りないぜ☆!

AMA

余しなさいよ!



「 書き足し用に数ページ残すが、検証する時間がないぜ☆! 続きは本番で☆!」



「時間があったとしても、ページ数が足りないけどな☆」



「 2行ぐらいしか 書くことのないチームから 24ページぐらい 買占めなさいよ!」



「なんで 時間も 余白も 足りなくなってしまうんだぜ、 トホホホホ…… …… ☆」

>>> Make unauthorized math!