

# パズル「ナンバープレイス」の難易度について

SATO, Kingo / 佐藤, 金吾

---

(出版者 / Publisher)

法政大学多摩研究報告編集委員会

(雑誌名 / Journal or Publication Title)

Hosei University Tama bulletin / 法政大学多摩研究報告

(巻 / Volume)

16

(開始ページ / Start Page)

41

(終了ページ / End Page)

96

(発行年 / Year)

2001-03-30

(URL)

<https://doi.org/10.15002/00003060>

# パズル「ナンバープレイス」の難易度について

佐藤金吾

On characteristic of difficult degree to solve Number-Place puzzles

Kingo SATO

## 1. はじめに

小論[1]において問題の難易度の客観的基準について言及した。しかし、イラストパズルでは、この点について若干の複雑さがあり、より簡単なナンバープレイスの難易度についてまず考えることにした。

ナンバープレイスとは、右図のような9×9のマス目を、表示されている数字（以下、表示数と呼ぶ）を手がかりに、つぎの2つのルールに従ってすべてのマスに数字を入れるパズルである：

1) タテ列の9マスと、ヨコ列の9マスに、1から9までの数字が1つずつ入る。

2) 太い線で区切られた3×3の9つのブロックの中にも、1から9までの数字が1つずつ入る。

（注意1.1）このルールからつぎのことがわかる。

3) 上記1)、2)いずれの場合も、同じ数字は2回使わない。

問題を解くとき、この3)の性質を上手に使うのがカギになる。

さて、以下の議論に必要なことを3点あげておく。

① タテ列、ヨコ列およびブロックの呼び方：

タテ列は左から順にタテ1列、タテ2列、……、タ

テ9列と、またヨコ列は上から順にヨコ1列、ヨコ2列、……、ヨコ9列と呼ぶ。

(例1)

			6	3	8			
5				2				3
	2	7				8	1	
		4				9		
		8	4		9	7		
		2				3		
	1	3				5	4	
8				7				9
			5	4	3			

ブロック 1	ブロック 2	ブロック 3
ブロック 4	ブロック 5	ブロック 6
ブロック 7	ブロック 8	ブロック 9

ブロックは右図のようにブロック1, ブロック2, …, ブロック9と呼ぶ。

② 実際の問題についての解の結果が必要なので、問題例として、手近にあった4冊の問題集 [2], [3], [4], [5] を用いた。

③ 本文中のプログラムには、言語として「構造化 BASIC98」を用いた。

## 2. ナンバープレイスの原理的解法

### 2.1 ナンバープレイスはつぎの方法で解ける：

(1) それぞれの空きマスについて、そのマスが属するタテ列, ヨコ列, ブロックの中に含まれる表示数を手がかりに、2つの方式：

[方式1] ダブらないで入れられる数字を調べ、1つだけが残った場合、残った数とそのマスに入る数字として決まる

[方式2] 表示数の中に8個の異なった数字が含まれていれば、1から9までの数の内でまだ使われていない数とそのマスに入る数字として決まる

を用いて表示数を確定する。

(2) それぞれの空きマスについて、そのマスが属するブロック内の核ブロック数やグループ確定数を手がかりに、方式1と方式2を用いて表示数を確定する。

(3) (1)、(2)によって表示数を増やしていき、すべてのマスが埋まるまで(1)、(2)の処理を繰り返す。

(注意2.1) 表示数だけを手がかりにした(1)と、増えた表示数をとり入れた繰り返しの(3)により、ほとんどの問題が解けるが、この(1)、(3)だけでは解けない問題がある。例えば、問題集 [2], [3] の対象になる178個の問題のうち、177個が (1)、(3) だけで解けるが、[2] の問題88番の1題のみ(2)の処理が必要となる。

そこで、(1)、(3)だけを使うものを処理1、さらに(2)を加えたものを処理2とよぶ。

(4) 2つの数字(数字対)が入る可能性をもつ空きマスを選び、その一方の数字を仮に入れて、上記処理(1)、(2)、(3)を行なう。このとき、

- ① 完成すれば、それが求める解である
- ② 途中で矛盾が出れば、仮の数字の対の他の数字がこのマスの表示数として決まる
- ③ 矛盾が出ないまま処理作業が途中でストップしたら、対で可能な新しい空きマスに代えて、上記処理を新規に行なう

という処理を加える(これを処理3とよぶ)。

(注意2.2) 処理2では解けない問題(例えば、[4]の問題70番)がある。

## 2.2 核ブロック数とグループ確定数について述べる。

(定義) あるブロック内において、タテ並びかヨコ並びである2つ以上の空きマスの列(間に表示数が挟まってもよい)を核ブロック空き列という。

核ブロック空き列において、その空きマスのどれかに入ることが分かっている数を核ブロック数という。

また、核ブロック空き列において、空きマスの数と同数の核ブロック数の組みが、個々の数字について入る空きマスを確認できない(組み全体としては決まる)とき、グループ確定数といわれる。

例えば、右図で、ブロック6における{2, 8}はグループ確定数であるが、ブロック7における{4, 6}はその空きマスがタテ並びでもヨコ並びでもないので、グループ確定数ではない。また、ブロック9における{2, 9}は\*印をつけた2つの空きマスをとればグループ確定数である。

さて、核ブロック数は、対応する空きマスのタテ(又はヨコ)並びを含むタテ(又はヨコ)列とその属するブロックに対して、

								8	
								2	
								3	
							1	6	
							3	9	
							5	4	7
2	5	1				9			
9		8				2			
	3	7				*		*	

- ① この数字は使用できない
- ② この数字が含まれている

という2つの情報(表示数を持つ重要な性質)を与え、方式1と方式2を若干拡張すれば表示数を見つけるための強力な手段となる。

なお、グループ確定数はその上に、

- ③ 対応する空きマスには、組みの数以外の数字は入れられない

という情報を与える。

ところで、核ブロック数およびグループ確定数には、つぎの2つのタイプがある：

[タイプ1] タテ列かヨコ列、あるいはブロックの確定した数字の残り(空きマスがあるブロックに集中してタテかヨコに並んだ場合)

[タイプ2] あるブロックのタテかヨコ並びの空きマスに注目したとき、ブロック内の条件(他の2列の計6個のマスに入れない)から決まる場合と、この並びを含むタテ列かヨコ列に関しての条件(他の2つのブロック部には入れない)から決まる場合がある。

そして、グループ確定数は、核ブロック数の組みがその縮小空きマスの個数と一致したものである。ただし、縮小空きマスとは、空きマスの内、対象となる数字グループのどれかが入れ

られないマスを除いたものである。

例えば、上記の例では、ブロック6の{2, 8}はタイプ1、ブロック9の{2, 9}はタイプ2である。

### 2.3 さて、原理的解法のコンピュータプログラムを示す。

このプログラムでは、 $9 \times 9$ のマス目に対応した配列に、そのマスに入れられない数字のリストを付け加えた配列変数：

$M$  (タテ列数,ヨコ列数,そのマスに入れようとする数字)

を使っている。

例えば、タテの3列目でヨコの5列目のマスに数字9が入れられないとき、 $M(3, 5, 9) = 1$  のように、入れられない場合に1とする。

```

100 REM *** Number Place ***
110 REM == genri_teki KAI ==
120 '
130 DIM M(9,9,9),HM(9,9,9)
140 DIM GTY(60),GRT(60),GBL(60),GSU(60,3),GSP(60,3)
150 DIM CTY(60),CRT(60),CBL(60),CSU(60,3)
160 DIM N(9),KS(9),SB(9),GR(9),SAKU(9),PK(9),A(3),SP(3)
170 CONSOLE „0: SCREEN 3: CLS 3
180 ' --- syoki_settei(1) ---
190 KAKUTEI=0
200 FOR I=1 TO 9:FOR J=1 TO 9
210 FOR K=0 TO 9: M(I,J,K)=0: NEXT K
220 NEXT J,I
230 ' --- data_yomi
240 ' & syoki_jyoho - input ---
250 FILE$="a:npprob.dat"
260 OPEN FILE$ FOR INPUT AS #1
270 WHILE NOT EOF(1)
280 INPUT #1,TA,YO,SU
290 GOSUB *KAKUTEI
300 WEND
310 ' --- suji_kakutei ---
320 ' syoki_settei(2)
330 SYORITYPE=1: OKAKUTEI=0: SSYORI3=0
340 ' --- loop_start ---
350 ESYORI3=0

```

```
360 WHILE ESYORI3=0
370 RLOOP=1
380 WHILE RLOOP=1
390 FEND=0
400 WHILE FEND=0
410 TUIKA=0
420 GOSUB *HOSIKI1
430 GOSUB *HOSIKI2
440 IF TUIKA=0 THEN FEND=1
450 WEND
460 IF (KAKUTEI=81)OR(KAKUTEI=OKAKUTEI) THEN
470 RLOOP=0
480 ELSE
490 ' -- syori_2 --
500 IF SYORITYPE<2 THEN SYORITYPE=2
510 OKAKUTEI=KAKUTEI
520 GOSUB *GROUP
530 GOSUB *INPJYHOH
540 END IF
550 WEND
560 IF KAKUTEI=81 THEN GOTO *KAIHYOJI
570 ' -- syori_3 --
580 SYORITYPE=3
590 IF SSYORI3=0 THEN
600 SSYORI3=1:LKSEN=0
610 ' syoki_settei
620 KOTA=1:KOYO=0
630 ' hikae_toru
640 HKAKUTEI=KAKUTEI
650 FOR I=1 TO 9:FOR J=1 TO 9
660 FOR K=0 TO 9: HM(I,J,K)=M(I,J,K): NEXT K
670 NEXT J,I
680 ELSE
690 ' atai_modosi
700 KAKUTEI=HKAKUTEI
710 FOR I=1 TO 9:FOR J=1 TO 9
720 FOR K=0 TO 9: M(I,J,K)=HM(I,J,K): NEXT K
```

```
730 NEXT J,I
740 END IF
750 ' kouho_sentak
760 IF LKSEN=0 THEN GOSUB *KOHOSSEN:LKSEN=1 ELSE LKSEN=0
770 ' kouho_toru
780 IF LKSEN=0 THEN SU=PK(2) ELSE SU=PK(1)
790 TA=KOTA:YO=KOYO:GOSUB *KAKUTEI
800 WEND
810 ' - - - kai_hyouji - - -
820 *KAIHYOJI
830 GOSUB *PRIN
840 LOCATE 5,22: PRINT "SYORITYPE=";SYORITYPE
850 END
860 :::::::::::::::::::::
1000 *PRIN
1010 XGS=168:XGL=384
1020 YGS=40:YGL=328
1030 ' - - waku_hyouji - -
1040 FOR YL=YGS TO YGL STEP 32
1050 LINE (XGS,YL) - (XGL,YL)
1060 IF (YL - YGS) MOD 96=0 THEN LINE (XGS,YL+1) - (XGL,YL+1)
1070 NEXT YL
1080 FOR XL=XGS TO XGL STEP 24
1090 LINE (XL,YGS) - (XL,YGL)
1100 IF (XL - XGS) MOD 72=0 THEN LINE (XL+1,YGS) - (XL+1,YGL)
1110 NEXT XL
1120 ' - - suu_hyouji - -
1130 FOR TA=1 TO 9:FOR YO=1 TO 9
1140 XHYJ=21+3*(YO - 1): YHYJ=3+2*(TA - 1)
1150 LOCATE XHYJ,YHYJ: PRINT M(TA,YO,0);
1160 NEXT YO,TA
1170 RETURN
1180 :::
1500 *KAKUTEI
1510 ' == par. TA,YO,SU ==
1520 KAKUTEI=KAKUTEI+1: TUIKA=1
1530 M(TA,YO,0)=SU
```

```
1540 FOR K9=1 TO 9: M(TA,YO,K9)=1: NEXT K9
1550 '
1560 FOR J9=1 TO 9: M(TA,J9,SU)=1: NEXT J9
1570 FOR I9=1 TO 9: M(I9,YO,SU)=1: NEXT I9
1580 B1=INT((TA - 1)/3)+1: B2=INT((YO - 1)/3)+1
1590 FOR I9=3*B1 - 2 TO 3*B1:FOR J9=3*B2 - 2 TO 3*B2
1600 M(I9,J9,SU)=1
1610 NEXT J9,I9
1620 '
1630 M(TA,YO,SU)=0
1640 RETURN
1650 :::
2000 *HOSIK1
2010 FOR TY=1 TO 3
2020 ' ty=1..gyou, ty=2..retu, ty=3..block
2030 IF (TY=1)OR(TY=2) THEN
2040 FOR I=1 TO 9: GOSUB *HSK1: NEXT I
2050 ELSE
2060 FOR II=1 TO 3:FOR JJ=1 TO 3
2070 GOSUB *HSK1
2080 NEXT JJ,II
2090 END IF
2100 NEXT TY
2110 RETURN
2120 :::
2500 *HSK1
2510 ' == par. TY,I(or II,JJ) ==
2520 FOR SU=1 TO 9
2530 PN=0: FITTI=0
2540 SELECT TY
2550 CASE 1,2
2560 FOR K9=1 TO 9
2570 IF TY=1 THEN MZ=M(I,K9,0):MS=M(I,K9,SU) ELSE MZ=M(K9,I,0): MS=M(K9,I,SU)
2580 IF MZ=SU THEN FITTI=1
2590 IF MS=1 THEN PN=PN+1 ELSE KN=K9
2600 NEXT K9
2610 IF TY=1 THEN TA=I:YO=KN ELSE TA=KN:YO=I
```



```

2620  CASE 3
2630  FOR I9=3*II - 2 TO 3*II:FOR J9=3*JJ - 2 TO 3*JJ
2640  IF M(I9,J9,0)=SU THEN FITTI=1
2650  IF M(I9,J9,SU)=1 THEN PN=PN+1 ELSE IN=I9:JN=J9
2660  NEXT J9,I9
2670  TA=IN:YO=JN
2680  END SELECT
2690  '
2700  IF (FITTI=0)AND(PN=8) THEN GOSUB *KAKUTEI
2710  NEXT SU
2720  RETURN
2730  :::
3000  *HOSIKI2
3010  FOR TA=1 TO 9:FOR YO=1 TO 9
3020  IF M(TA,YO,0)=0 THEN
3030  PN=0
3040  FOR K9=1 TO 9
3050  IF M(TA,YO,K9)=1 THEN PN=PN+1 ELSE KN=K9
3060  NEXT K9
3070  IF PN=8 THEN SU=KN:GOSUB *KAKUTEI
3080  END IF
3090  NEXT YO,TA
3100  RETURN
3110  :::
4000  *GROUP
4010  GN=0:CBN=0
4020  FOR TY=1 TO 2
4030  ' ty=1..gyou, ty=2..retu
4040  FOR I=1 TO 9:FOR BL=1 TO 3
4050  GOSUB *AKISU
4060  IF AKI>=2 THEN
4070  GOSUB *KANOUSU
4080  IF KSN=AKI THEN
4090  ' -- type_1 --
4100  ' kiroku
4110  GN=GN+1
4120  GTY(GN)=TY:GRT(GN)=I:GBL(GN)=BL:GSU(GN,0)=KSN

```

```
4130     FOR K9=1 TO KSN: GSP(GN,K9)=KS(K9): NEXT K9
4140     GSP(GN,0)=KSN
4150     FOR K9=1 TO KSN: GSP(GN,K9)=A(K9): NEXT K9
4160     ELSE
4170     ' -- type_2 --
4180     GOSUB *SUBGROUP
4190     END IF
4200     END IF
4210     NEXT BL,I
4220     NEXT TY
4230     RETURN
4240     :::
4300     *AKISU
4310     AKI=0
4320     FOR J=3*BL - 2 TO 3*BL
4330     IF TY=1 THEN MS=M(I,J,0) ELSE MS=M(J,I,0)
4340     IF MS=0 THEN AKI=AKI+1:A(AKI)=J
4350     NEXT J
4360     RETURN
4370     ::
4400     *KANOUSU
4410     FOR I9=1 TO 9: N(I9)=0: NEXT I9
4420     '
4430     FOR A9=1 TO AKI
4440     FOR S9=1 TO 9
4450     IF TY=1 THEN MS=M(I,A(A9),S9) ELSE MS=M(A(A9),I,S9)
4460     IF MS=0 THEN N(S9)=1
4470     NEXT S9
4480     NEXT A9
4490     '
4500     KSN=0
4510     FOR I9=1 TO 9
4520     IF N(I9)=1 THEN KSN=KSN+1:KS(KSN)=I9
4530     NEXT I9
4540     RETURN
4550     ::
5000     *SUBGROUP
```

```
5010 ' -- jyunbi --
5020 SBN=0
5030 FOR K9=1 TO KSN
5040 SU=KS(K9)
5050 ' -- block_nai_cond. --
5060 NSN=0: II=INT((I - 1)/3)+1
5070 FOR I9=3*II - 2 TO 3*II
5080 IF I9<>I THEN
5090 FOR J9=3*BL - 2 TO 3*BL
5100 IF TY=1 THEN MS=M(I9,J9,SU) ELSE MS=M(J9,I9,SU)
5110 IF MS=1 THEN NSN=NSN+1
5120 NEXT J9
5130 END IF
5140 NEXT I9
5150 IF NSN=6 THEN
5160 SBN=SBN+1:SB(SBN)=SU
5170 ELSE
5180 ' -- retu_gai_cond. --
5190 FSUBG=1
5200 FOR BN=1 TO 3
5210 IF BN<>BL THEN
5220 FOR J9=3*BN - 2 TO 3*BN
5230 IF TY=1 THEN MS=M(I,J9,SU) ELSE MS=M(J9,I,SU)
5240 IF MS=0 THEN FSUBG=0
5250 NEXT J9
5260 END IF
5270 NEXT BN
5280 IF FSUBG=1 THEN SBN=SBN+1:SB(SBN)=SU
5290 END IF
5300 NEXT K9
5310 ' -- hantei --
5320 IF (1<=SBN)AND(SBN<=AKI) THEN
5330 CHN=0
5340 FOR J9=3*BL - 2 TO 3*BL
5350 FCH=1
5360 FOR K9=1 TO SBN
5370 SU=SB(K9)
```

```
5380     IF TY=1 THEN MS=M(I,J9,SU) ELSE MS=M(J9,I,SU)
5390     IF MS=0 THEN FCH=0
5400     NEXT K9
5410     IF FCH=1 THEN CHN=CHN+1: SAKU(CHN)=J9
5420     NEXT J9
5430     IF SBN=3 - CHN THEN
5440       ' group_kakusu
5450       GN=GN+1
5460       GTY(GN)=TY:GRT(GN)=I:GBL(GN)=BL:GSU(GN,0)=SBN
5470       FOR K9=1 TO SBN: GSU(GN,K9)=SB(K9): NEXT K9
5480       GOSUB *GSPACE
5490       GSP(GN,0)=GSPN
5500       FOR K9=1 TO GSPN: GSP(GN,K9)=SP(K9): NEXT K9
5510     ELSE
5520       ' kaku_blocksu
5530       CBN=CBN+1
5540       CTY(CBN)=TY:CRT(CBN)=I:CBL(CBN)=BL:CSU(CBN,0)=SBN
5550       FOR K9=1 TO SBN: CSU(CBN,K9)=SB(K9): NEXT K9
5560     END IF
5570 END IF
5580 RETURN
5590 :::
5600 *GSPACE
5610 FOR I9=1 TO 9: N(I9)=0: NEXT I9
5620 FOR I9=1 TO AKI: N(A(I9))=1: NEXT I9
5630 FOR I9=1 TO CHN: N(SAKU(I9))=0: NEXT I9
5640 '
5650 GSPN=0
5660 FOR I9=1 TO 9
5670   IF N(I9)=1 THEN GSPN=GSPN+1:SP(GSPN)=I9
5680 NEXT I9
5690 RETURN
5700 ::::
6000 *INPJYOHO
6010 ' -- group_kakusu --
6020 FOR U=1 TO GN
6030 TY=GTU(U):I=GRT(U):BL=GBL(U):NN=GSU(U,0):SPN=GSP(U,0)
```

```
6040 ' gyou(or retu)
6050 FOR K=1 TO NN
6060   FOR BN=1 TO 3
6070     IF BN<>BL THEN
6080       FOR J9=3*BN - 2 TO 3*BN
6090         IF TY=1 THEN M(I,J9,GSU(U,K))=1 ELSE M(J9,I,GSU(U,K))=1
6100       NEXT J9
6110     END IF
6120   NEXT BN
6130 NEXT K
6140 ' block
6150 B1=INT((I - 1)/3)+1
6160 FOR K=1 TO NN
6170   FOR I9=3*B1 - 2 TO 3*B1
6180     IF I9<>I THEN
6190       FOR J9=3*BL - 2 TO 3*BL
6200         IF TY=1 THEN M(I9,J9,GSU(U,K))=1 ELSE M(J9,I9,GSU(U,K))=1
6210       NEXT J9
6220     END IF
6230   NEXT I9
6240 NEXT K
6250 '
6260 GOSUB *GRMAKE
6270 FOR K=1 TO SPN
6280   PJ=GSP(U,K)
6290   FOR I9=1 TO GRN
6300     IF TY=1 THEN M(I,PJ,GR(I9))=1 ELSE M(PJ,I,GR(I9))=1
6310   NEXT I9
6320 NEXT K
6330 NEXT U
6340 ' -- kaku_blocksu --
6350 FOR U=1 TO CBN
6360   TY=CTY(U):I=CRT(U):BL=CBL(U):NN=CSU(U,0)
6370 ' gyou(or retu)
6380 FOR K=1 TO NN
6390   FOR BN=1 TO 3
6400     IF BN<>BL THEN
```

```
6410   FOR J9=3*BN - 2 TO 3*BN
6420     IF TY=1 THEN M(I,J9,CSU(U,K))=1 ELSE M(J9,I,CSU(U,K))=1
6430   NEXT J9
6440   END IF
6450   NEXT BN
6460   NEXT K
6470   ' block
6480   B1=INT((I - 1)/3)+1
6490   FOR K=1 TO NN
6500     FOR I9=3*B1 - 2 TO 3*B1
6510       IF I9<>I THEN
6520         FOR J9=3*BL - 2 TO 3*BL
6530           IF TY=1 THEN M(I9,J9,CSU(U,K))=1 ELSE M(J9,I9,CSU(U,K))=1
6540         NEXT J9
6550       END IF
6560     NEXT I9
6570   NEXT K
6580   NEXT U
6590   RETURN
6600   :::
6700   *GRMAKE
6710   FOR I9=1 TO 9: N(I9)=0: NEXT I9
6720   FOR I9=1 TO NN: N(GSU(U,I9))=1: NEXT I9
6730   GRN=0
6740   FOR I9=1 TO 9
6750     IF N(I9)=0 THEN GRN=GRN+1: GR(GRN)=I9
6760   NEXT I9
6770   RETURN
6780   :::
7000   *KOHSEN
7010   IF KOYO<9 THEN STA=KOTA:SYO=KOYO+1 ELSE STA=KOTA+1:SYO=1
7020   '
7030   FOR TA=STA TO 9
7040     IF TA>STA THEN SYO=1
7050     FOR YO=SYO TO 9
7060       IF M(TA,YO,0)=0 THEN
7070         ' koho_sagasu
```

```

7080  PUTN=0
7090  FOR K9=1 TO 9
7100    IF M(TA,YO,K9)=0 THEN PUTN=PUTN+1:PK(PUTN)=K9
7110  NEXT K9
7120  IF PUTN=2 THEN KOTA=TA:KOYO=YO: RETURN
7130  END IF
7140  NEXT YO
7150  NEXT TA
7160  RETURN

```

### 3. 難易度の基準化 (1)

さて、原理的解法を踏まえた上で、難易度の基準化を考える。

この時、重要な点は人間がやれる方法、標準的にやる方法を分析し、それを手がかりにすることである。原理的解法は、方式1,方式2をすべての場合に総当たりのに行なっているが、人間にはそんなことはまず不可能であり、誰もやりはしないので、人間のやる方法に根ざしてつけられた難易度についての基準化には役立たない。

だから、総当たりのではなく、限られたマスに絞り込み、そこだけに集中して適用する方法を手がかりにする。それが人間のやり方だからであり、ちょっとしたメモや書き込み程度で出来る方法でもある。

#### 3.1 基本になる3つの技法

[技法1] あるタテ列かヨコ列の9マス、またはあるブロックの9マスにおいて、8個のマスで表示数が決まっている時、残りの空きマスには1から9までの数の内でまだ使われてない数字が入る。

[技法2] タテないしヨコの3つのマス列の組：

①1列, 2列, 3列 か ②4列, 5列, 6列 か ③7列, 8列, 9列

において、表示数が2つある数字（以下2ペア数と呼ぶ）に注目して、つぎの処理を行なう：

2ペア数を含まないブロックで、2ペア数を含まないマス列の一部である核ブロック空き列を考え（この時、2ペア数はこの核ブロック数になる）、2ペア数がこの空きマス列のどれかの表示数になれるかを、関係するまわりの表字数を手がかりに調べる（空きマスが1個なら自動的に表示数と決まる）。

ダブらないマスがひとつだけ残れば（他のマスはダブって入れられない）、2ペア数はそのマスの表示数に決まる。

[技法3] あるタテ列かヨコ列の9マス、またはあるブロックの9マスにおいて、その空きマスを考える。この空きマスだけに限定して、方式1, 方式2を適用して表示数を確定する。この場合、空きマスの数を  $n$  個とする時、 $n$ -空きマス技法と呼ぶ。

(実例) 例1のナンバープレイスでは、右の図1の○ (図1)

で囲まれた11個の数字が技法2により表示数として決まる。例えば、ヨコ列の7列, 8列, 9列の3つの組において、数字3は2ペア数である。数字3が含まれないブロック9のヨコ8列の一部の核ブロック空き列(2つの空きマスあり)において、タテ7列に数3があるため、この核ブロック空き列の最初のマス(左からみて)に数字3が入れられず、従って2番目のマスに数字3が決まる。

④			6	3	8			
5	⑧			2		④		3
③	2	7	⑨	⑤	④	8	1	⑥
		4	③			9		
	③	8	4		9	7		
		2				3		④
	1	3				5	4	
8	④	⑤		7			③	9
			5	4	3			

また、ヨコ3列目の3つの空きマスにおいて、技法3より、左から4番目のマスに数9、5番目のマスに数5、9番目のマスに数6が決まる(決まる順番は数6,数5,そして数9の順)。数6が決まるのは、左から4番目と5番目の空きマスに数6がダブりのために入れられず、残った9番目の空きマスに数6が入るからである。

(注意3.1) 技法1は1-空きマス技法と一致する。明らかなように、空きマスの個数が多くなるほど、技法3の適用が急激に難しくなる。この小論では、この難しさを難易度分析の核にすえている。

### 3.2 3つの技法を用いた解法

1) 技法2をタテ列の組の3つとヨコ列の組の3つ、計6つについて組織的に完璧に行ない、2ペア数で表示数になるものをすべてとり出す。

2) 1)が終わったら、各タテ列,各ヨコ列,そして各ブロック(全部で対象が27となる)について含まれる表示数を数え、空きマスの一番少ない列かブロックから始めて順に空きマスの多いものへと技法3を適用していく。そして、一つでも表示数が確定したら、1)にもどり新しく得た表示数を加えて技法2を用いる。

(図2)

4	⑨	①	6	3	8	②		
5	8	⑥		2		4	⑨	3
3	2	7	9	5	4	8	1	6
		4	3			9		
	3	8	4		9	7		
⑨		2				3		4
	1	3	⑧	⑨		5	4	
8	4	5		7			3	9
		⑨	5	4	3			

3) すべてのマスの表示数が決まるまで、1), 2)を繰り返す。

(実例) 例1のナンバープレイスを考える。



1)により図1の11個の表示数が得られる。つぎに空きマス調べると、タテ3列、タテ7列、ヨコ3列、ヨコ8列、ブロック1の計5つが最小の空きマス3を持つ。そこで2)を、例えばヨコ3列に適用すると、図1の数9, 数5, 数6が決まる。

1)に戻って、技法2より2つの表示数6, 9が決まる(右の図2の○で囲ったもの)。2)に進んで、最小の空きマス2をもつタテ3列に技法3を適用すると、上から1番目のマスが数1, 9番目のマスが数9となる(図2の□で囲ったもの)。すると、技法1よりブロック1で数9が決まり(図2の◇で囲ったもの)、また1)より2つの表示数が得られる(図2の◎で囲ったもの)。2)に進んで、ブロック2は最小の空きマス2をもつが、ここから表示数は得られないので、空きマス3のものに処理を移す。タテ7列で方式2より上から一番目のマスが数2と決まる(図2の□で囲ったもの)。そこで、1)に戻るが新しい表示数は得られず、2)に進む。最小空きマス2から始めて、空きマス3へと移るが、いずれも表示数は見つからず、空きマス4に移る。すると、ブロック8で、方式1より数8が決まる(図2の□で囲ったもの)。

同様に続けると、この問題は4-空きマス技法まで使って解ける。

(注意3.2) 上の方法でほとんどの問題が解ける。実際、[2], [3]の対象となる178題のうち、177題が解ける。

### 3.3 上の解法に従ったコンピュータプログラムを示す。

このプログラムの特徴は、前の原理的プログラムと違い、 $9 \times 9$ のマス目に対応した配列に、そのマスに入れられない数字のリストを付け加えていないことである。

このリストは原理的・総当たりに処理する時に適している。リストをつけないほうが、プログラムは多少厄介になるが人間のやる方法にマッチしている。

```

100 REM *** Number Place ***
110 REM == nanido_kijyun(1) ==
120 DIM M(9,9),P(9),PI(9,3),A(9),AA(9,2),GS(9),N(9)
130 CONSOLE ,,0,1: SCREEN 3: CLS 3
140 ' --- syoki_settei ---
150 KAKUTEI=0
160 FOR I=1 TO 9:FOR J=1 TO 9
170 M(I,J)=0
180 NEXT J,I
190 ' --- data_yomi
200 ' & syoki_jyoho - input ---
210 FILE$="a:npprob.dat"

```

```
220 OPEN FILE$ FOR INPUT AS #1
230 WHILE NOT EOF(1)
240 INPUT #1,TA,YO,SU
250 GOSUB *KAKUTEI
260 WEND
270 ' --- su_kakutei ---
280 FEND=0: MAKIN=0
290 WHILE FEND=0
300 TUIKA=0
310 FOR NAKI=1 TO 6
320 GOSUB *GIHOU2
330 IF TUIKA=1 THEN GOTO *ENAKI
340 GOSUB *GIHOU3
350 IF TUIKA=1 THEN GOTO *ENAKI
360 NEXT NAKI
370 *ENAKI
380 IF (KAKUTEI=81)OR(TUIKA=0) THEN FEND=1
390 WEND
400 ' --- kai_hyoji ---
410 GOSUB *PRIN
420 LOCATE 5,22: PRINT "N - AKI=";MAKIN
430 END
440 :::::::::::
1000 *PRIN
1010 XGS=168:XGL=384
1020 YGS=40:YGL=328
1030 ' -- waku_hyouji --
1040 FOR YL=YGS TO YGL STEP 32
1050 LINE (XGS,YL) - (XGL,YL)
1060 IF (YL - YGS) MOD 96=0 THEN LINE (XGS,YL+1) - (XGL,YL+1)
1070 NEXT YL
1080 FOR XL=XGS TO XGL STEP 24
1090 LINE (XL,YGS) - (XL,YGL)
1100 IF (XL - XGS) MOD 72=0 THEN LINE (XL+1,YGS) - (XL+1,YGL)
1110 NEXT XL
1120 ' -- suu_hyouji --
1130 FOR TA=1 TO 9:FOR YO=1 TO 9
```

```
1140 IF M(TA,YO)>0 THEN
1150 XHYJ=21+3*(YO - 1): YHYJ=3+2*(TA - 1)
1160 LOCATE XHYJ,YHYJ: PRINT M(TA,YO);
1170 END IF
1180 NEXT YO,TA
1190 RETURN
1200 :::
1500 *KAKUTEI
1510 ' == par. TA,YO,SU ==
1520 KAKUTEI=KAKUTEI+1: TUIKA=1
1530 M(TA,YO)=SU
1540 RETURN
1550 ::::::::::::::::::::
2000 *HOSIKI2
2010 ' == par. TA,YO ==
2020 FOR U=1 TO 9: N(U)=0: NEXT U
2030 '
2040 FOR U=1 TO 9: N(M(TA,U))=1: NEXT U
2050 FOR U=1 TO 9: N(M(U,YO))=1: NEXT U
2060 B1=INT((TA - 1)/3)+1:B2=INT((YO - 1)/3)+1
2070 FOR U1=3*B1 - 2 TO 3*B1:FOR U2=3*B2 - 2 TO 3*B2
2080 N(M(U1,U2))=1
2090 NEXT U2,U1
2100 '
2110 PN=0
2120 FOR U=1 TO 9
2130 IF N(U)=1 THEN PN=PN+1 ELSE KN=U
2140 NEXT U
2150 IF PN=8 THEN
2160 SU=KN: GOSUB *KAKUTEI
2170 END IF
2180 RETURN
2190 :::
3000 *GIHOU2
3010 REP=1
3020 WHILE REP=1
3030 REP=0
```

```
3040 FOR TY=1 TO 2
3050 ' ty=1 -- tate ty=2 -- yoko
3060 FOR II=1 TO 3
3070 GOSUB *TECH2
3080 NEXT II
3090 NEXT TY
3100 WEND
3110 RETURN
3120 :::
3500 *TECH2
3510 ' == par. TY,II ==
3520 ' syoki_ka
3530 FOR I9=1 TO 9
3540 P(I9)=0
3550 FOR J9=0 TO 3: PI(I9,J9)=0:PJ(I9,J9)=0: NEXT J9
3560 NEXT I9
3570 '
3580 FOR I9=3*II - 2 TO 3*II
3590 FOR J9=1 TO 9
3600 IF TY=1 THEN MS=M(I9,J9) ELSE MS=M(J9,I9)
3610 IF MS>0 THEN
3620 P(MS)=P(MS)+1
3630 PI(MS,P(MS))=I9:PJ(MS,P(MS))=INT((J9 - 1)/3)+1
3640 END IF
3650 NEXT J9
3660 NEXT I9
3670 '
3680 FOR SU=1 TO 9
3690 IF P(SU)=2 THEN
3700 GRI=9*(II - 1)+6 - (PI(SU,1)+PI(SU,2)):GRJ=6 - (PJ(SU,1)+PJ(SU,2))
3710 PN=0
3720 FOR J9=3*GRJ - 2 TO 3*GRJ
3730 IF TY=1 THEN TA=GRI:YO=J9 ELSE TA=J9:YO=GRI
3740 IF M(TA,YO)>0 THEN
3750 PN=PN+1
3760 ELSE
3770 FOR J8=1 TO 9
```

```
3780     IF TY=1 THEN MS=M(J8,YO) ELSE MS=M(TA,J8)
3790     IF MS=SU THEN PN=PN+1:GOTO *EXJ8
3800     NEXT J8
3810     JN=J9
3820     *EXJ8
3830     END IF
3840     NEXT J9
3850     '
3860     IF PN=2 THEN
3870         IF TY=1 THEN TA=GRI:YO=JN ELSE TA=JN:YO=GRI
3880         GOSUB *KAKUTEI
3890         REP=1
3900     END IF
3910     END IF
3920     NEXT SU
3930     RETURN
3940     :::
4000 *GIHOU3
4010 IF NAKI>MAKIN THEN MAKIN=NAKI
4020 ' retu_syori
4030 FOR TTY=1 TO 2
4040 ' tty=1..yoko tty=2..tate
4050 FOR I=1 TO 9
4060     GOSUB *TECH3
4070     IF TUIKA=1 THEN RETURN
4080 NEXT I
4090 NEXT TTY
4100 ' block_syori
4110 FOR BI=1 TO 3:FOR BJ=1 TO 3
4120     TTY=3
4130     GOSUB *TECH3
4140     IF TUIKA=1 THEN RETURN
4150 NEXT BJ,BI
4160 RETURN
4170 :::
5000 *TECH3
5010 ' == par. TTY,I(or BI,BJ) ==
```

```
5020 ' -- aki_check --
5030 GOSUB *AKICH
5040 ' -- kakutei_syori --
5050 IF (1<=AKI)AND(AKI<=NAKI) THEN
5060 ' -- hosiki_1 --
5070 FOR S9=1 TO AKI
5080 SU=GS(S9)
5090 PN=0
5100 SELECT TTY
5110 CASE 1,2
5120 FOR J9=1 TO AKI
5130 IF TTY=1 THEN TA=I:YO=A(J9) ELSE TA=A(J9):YO=I
5140 GOSUB *ITTICH
5150 IF FITTI=1 THEN PN=PN+1 ELSE JN=A(J9)
5160 NEXT J9
5170 IF TTY=1 THEN TA=I:YO=JN ELSE TA=JN:YO=I
5180 CASE 3
5190 FOR K9=1 TO AKI
5200 TA=AA(K9,1):YO=AA(K9,2): GOSUB *ITTICH
5210 IF FITTI=1 THEN PN=PN+1 ELSE IN=AA(K9,1):JN=AA(K9,2)
5220 NEXT K9
5230 TA=IN:YO=JN
5240 END SELECT
5250 '
5260 IF PN=AKI - 1 THEN GOSUB *KAKUTEI:RETURN
5270 NEXT S9
5280 ' -- hosiki_2 --
5290 FOR K9=1 TO AKI
5300 IF TTY=1 THEN TA=I:YO=A(K9)
5310 IF TTY=2 THEN TA=A(K9):YO=I
5320 IF TTY=3 THEN TA=AA(K9,1):YO=AA(K9,2)
5330 GOSUB *HOSIKI2
5340 IF TUIKA=1 THEN RETURN
5350 NEXT K9
5360 END IF
5370 RETURN
5380 ::
```

```
5500 *AKICH
5510 ' == par. TTY ==
5520 FOR I9=1 TO 9: N(I9)=0: NEXT I9
5530 '
5540 AKI=0
5550 SELECT TTY
5560 CASE 1,2
5570     FOR J9=1 TO 9
5580         IF TTY=1 THEN MS=M(1,J9) ELSE MS=M(J9,1)
5590         IF MS=0 THEN AKI=AKI+1:A(AKI)=J9 ELSE N(MS)=1
5600     NEXT J9
5610 CASE 3
5620     FOR I9=3*B1 - 2 TO 3*B1:FOR J9=3*B2 - 2 TO 3*B2
5630         IF M(I9,J9)=0 THEN AKI=AKI+1:AA(AKI,1)=I9:AA(AKI,2)=J9 ELSE N(M(I9,J9))=1
5640     NEXT J9,I9
5650 END SELECT
5660 '
5670 GSN=0
5680 FOR I9=1 TO 9
5690     IF N(I9)=0 THEN GSN=GSN+1:GS(GSN)=I9
5700 NEXT I9
5710 RETURN
5720 ::
5800 *ITTICH
5810 ' == par. TA,YO,SU ==
5820 FITTI=0
5830 FOR U9=1 TO 9
5840     IF M(TA,U9)=SU THEN FITTI=1
5850     IF M(U9,YO)=SU THEN FITTI=1
5860 NEXT U9
5870 B1=INT((TA - 1)/3)+1:B2=INT((YO - 1)/3)+1
5880 FOR U9=3*B1 - 2 TO 3*B1:FOR U8=3*B2 - 2 TO 3*B2
5890     IF M(U9,U8)=SU THEN FITTI=1
5900 NEXT U8,U9
5910 RETURN
```

### 3.4 178題の解法結果による難易度の基準

上記プログラムによる、[2], [3]の対象となる178題の解法結果を示す。

問題の難易度はレベル1からレベル3までであり、レベルが高いほど難易度も高くなる。

表の横列には解くのに必要とした $n$ -空きマス技法の $n$ の最大値をとっている。

ただし、この解法では解けない1題は6-空きマス技法以上に入れた。

	1-空きマス 技法まで	2-空きマス 技法まで	3-空きマス 技法まで	4-空きマス 技法まで	5-空きマス 技法まで	6-空きマス 技法以上	計
レベル1	3	4	6	1	0	0	14
レベル2	0	10	33	32	4	0	79
レベル3	0	1	8	37	34	5	85
計	3	15	47	70	43	5	178

この表から必要とした $n$ -空きマス技法の $n$ の値とレベルの間にはかなりの関連性があることが予想される。実際、その相関係数を計算すると、

$$r = 0.633$$

となる。従って、以下のような難易度の基準が考えられる。

[基準A] 解法に必要な $n$ -空きマス技法の $n$ の値により、

$1 \leq n \leq 3$  ならレベル1,  $2 \leq n \leq 4$  ならレベル2,  $3 \leq n$  ならレベル3とする。

従って、 $n \geq 5$  ならレベル3である。

(注意3.3) しかし、この基準は $n$ の値でのダブりがあり(例えば、 $n=3$ から一意にレベルが決まらない)、より精密な基準を検討する必要がある。

## 4. 難易度の基準化(2)

### 4.1 基準化のための評価値について

(1) 81個のそれぞれのマスに、その表示数を決めるのに用いた技法に応じたポイントを付ける。そして、マス全体にわたるポイントの平均値を評価値としてとる。

(2) 使用する技法は、今までにあげた技法1から技法3までの3つとする。

(3) 技法に応じたポイントをつぎのようにとる。

- |               |      |   |      |
|---------------|------|---|------|
| ① はじめの確定数     | .... | 0 | ポイント |
| ② 技法1によって確定する | .... | 2 | "    |
| ③ 技法2 " "     | .... | 5 | "    |



④	技法3で2-空きマス技法による	....	4	”
⑤	” 3- ” ”	....	11.5	”
⑥	” 4- ” ”	....	17	”
⑦	” 5- ” ”	....	24	”
⑧	” 6- ” ”	....	32	”

(4) 解いていく順番、すなわち、使用する技法の順番は、ポイントが低い技法の順に行なう。例えば、技法1と技法2では、ポイントの低い技法1を先に行なう。

(5)  $n$ -空きマス技法の対象になるタテ列、ヨコ列、ブロックが複数あるとき、どれを優先するかをあらかじめ決めておかなければならない。しかし、優先順位の決め方によって評価値が変化するので、評価値の計算はつぎのようにする：

タテ列→ヨコ列→ブロック，タテ列→ブロック→ヨコ列，・・・，ブロック→ヨコ列→タテ列とした6通りの優先順位のものについてそれぞれ計算し、その最小値を評価値とする（人間がやる場合は問題の特殊性を利用するのでそれを少しでも反映させる）。

(6) 解法手順は、3.2で示したように最初に技法2を用いて表示数、すなわち、手がかりとなる数をまず増やす。この操作に伴う評価値の計算をつぎのようにする：

① (1)とは矛盾するが、ここで求めた表示数にはポイントとして0を与える、

② そのかわり全体のポイントに7.5を加算する。何故なら、表示数が見つかる見つからないにかかわらず、この技法2の操作には同じ手間が共通にかかるからである。

(7) 3節で定めた基準Aを以下のように用いる。

① レベル3の2-空きマス技法までの1題は対象から除く。

② レベル1の4-空きマス技法までの1題と、レベル2の5-空きマス技法までの4題は対象から除く。

つまり、やさしすぎたり、問題の持つ特殊性が強すぎるものは対象から除く。

③ 5-空きマス技法まで（及びそれ以上）必要とするものは、評価値の値に関係なくレベル3とする。

#### 4.2 ポイントの決め方

ポイントの決め方は要する手間に準ずる。すなわち、各技法の処理の仕方を分析し、その作業数に応じたポイントを総和する。

特に、数の集団からある数を見つける時、1から9までの数が順に書かれた用紙にチェックを入れるという操作を基本にし、その操作をポイント1とする。

(1) 技法1のポイント

空きマスに入る数を8個の数の集団から見つければよい。1番目から8番目までのそれぞれの数

に応じて、上記用紙にチェックを入れるので、8Pとなる。

(2) 技法2のポイント

(イ) 最初の場合：

① 2ペア数を見つける . . . . 18P

・1列に平均4.5個の表示数がある(0~9の平均)。上記用紙1枚にチェックを入れていくと、 $4.5 \times 3 = 13.5P$ 。

・チェックが2つ付いたものを探す(.5P)と、 $.5 \times 9 = 4.5P$

② 2ペア数が確定数となるかの調べ . . . . 11P

・該当ブロックを見つける(マス消し作業は.25P) . . . .  $(27-3) \times .25 = 6P$

・空きマスは1個,2個,3個の場合があり、それぞれの割合は1/3ずつとする。

空きマスが2個のとき、周りの表示数に一致数があるか調べる . . . . 平均3個の表示数があるとして、 $3 \times 2 = 6P$ 。

空きマスが3個のとき、同様にして、 $3 \times 3 = 9P$ 。

従って、 $1/3 \times (0+6+9) = 5P$ 。

以上を総和して、 $18+11=29P$ 。

(イ) 途中の場合：

① 2ペア数を見つける(ある並びのある数について行なう)

1列に平均4.5個の表示数があり、上記用紙へのチェック数は、 $4.5 \times 2 = 9P$ 。

② 2ペア数が確定数となるかの調べ . . . . 11P

以上を総計して、 $9+11=20P$ 。

(3) 2-空きマス技法のポイント

① 空きマスに入る2個の数を7個の表示数からを見つける . . . . 7P

② 上記用紙1枚を用意し、空きマスに入る2個の数に印をつける . . . . 2P

③ 空きマスの周り(列とブロックの両方)の表示数の中にこの2個の数があるかどうかを調べる。

・周りの列とブロックには平均して5個の表示数がある。上記用紙に、それぞれの空きマスに応じた数チェックを行ない、その過程で印がついた箇所にチェックが入れば終わる . . . . 一番早くて1P,一番遅くて10P

・該当数がある場合はこの半分ですむとし、該当数があるかないかの割合を .5, .5とすれば、 $5 \times .5 + 10 \times .5 = 7.5P$

以上から、求めるポイントは16.5Pとなる。

(4) 3-空きマス技法のポイント

① 空きマスに入る3個の数を6個の表示数から見つける . . . . 6 P

② 上記用紙2枚を用意し、空きマスに入る3個の数に印をつける . . . . 3 Pと3 P

③ 空きマスの周り（列とブロックの両方）の表示数をこの用紙にチェックして入れていく。  
まず1枚目の用紙で行ない、それで表示数が求まらない時は2枚目に移る。

(イ) 1枚目の用紙の印がついた3個の内の1個が2回チェックされる場合：

周りの列とブロックには平均して5.25個の表示数がある（全体で2個～40個）。

一番早くて6.25回、一番遅くて15.75回のチェックが行なわれ、また、2回をカウント（.5 P）. . . .

早くて.5 P,遅くて1.5 P。

(ロ) 2枚目の用紙が、ある空きマスのチェックで印がついた3箇所の中の2箇所にチェックが入る場合：

周りの列とブロックには平均して6個の表示数がある（2個～10個の平均）。

一つの空きマスのチェックがすむ . . . . 6 P,

3箇所のチェックを確認 . . . . 早くて2 P,遅くて3 P。

・該当数があるかないかの割合を  $1/3, 2/3$  とし、その  $1/3$  の内の  $3/4$  が (イ) 型であると  
する。

該当数がない場合のポイントは、 $6+3+3+15.75+1.5+3\times(6+3)=56.25$  P。

(イ) 型の該当数がある場合のポイントは、 $6+3+11+1=21$  P。

(ロ) 型の該当数がある場合のポイントは、 $6+6+17.25+3\times(6+3)/2=42.75$  P。

以上から、求めるポイントは、

$$1/3 \times 3/4 \times 21 + 1/3 \times 1/4 \times 42.75 + 2/3 \times 56.25 = 46.3 \text{ P.}$$

(5) 4-空きマス技法のポイント

① 空きマスに入る4個の数を5個の表示数から見つける . . . . 5 P

② 上記用紙2枚を用意し、空きマスに入る4個の数に印をつける . . . . 4 Pと4 P

③ 空きマスの周り（列とブロックの両方）の表示数をこの用紙にチェックして入れていく。  
まず1枚目の用紙で行ない、それで表示数が求まらない時は2枚目に移る。

(イ) 1枚目の用紙の印がついた4個の内の1個が3回チェックされる場合：

周りの列とブロックには平均して5.4個の表示数がある（全体で3個～40個）。

一番早くて11.8回、一番遅くて21.5回のチェックが行なわれ、また、

3回をカウント（1 P）. . . . 早くて1 P,遅くて4 P。

(ロ) 2枚目の用紙が、ある空きマスのチェックで印がついた4箇所の中の3箇所にチェックが入る場合：

周りの列とブロックには平均して6.5個の表示数がある。

一つの空きマスのチェックがすむ . . . . 6.5 P,

4箇所のチェックを確認 . . . . 早くて3 P, 遅くて4 P。

・該当数があるかないかの割合を  $1/4, 3/4$  とし、その  $1/4$  の内の  $4/5$  が (イ) 型であると  
する。

該当数がない場合のポイントは、 $5+4+4+21.5+4+4 \times (6.5+4) = 80.5 P$

(イ) 型の該当数がある場合のポイントは、 $5+4+16.65+2.5 = 28.15 P$

(ロ) 型の該当数がある場合のポイントは、 $5+8+25.5+4 \times (6.5+4)/2 = 59.5 P$ 。

以上から、求めるポイントは、

$$1/4 \times 4/5 \times 28.15 + 1/4 \times 1/5 \times 59.5 + 3/4 \times 80.5 = 69 P。$$

(6) 5-空きマス技法のポイント

① 空きマスに入る5個の数を4個の表示数から見つける . . . . 4 P

② 上記用紙2枚を用意し、空きマスに入る5個の数に印をつける . . . . 5 P と 5 P

③ 空きマスの周り (列とブロックの両方) の表示数をこの用紙にチェックして入れていく。

まず1枚目の用紙で行ない、それで表示数が求まらない時は2枚目に移る。

(イ) 1枚目の用紙の印がついた5個の内の1個が4回チェックされる場合：

周りの列とブロックには平均して5.5個の表示数がある。

一番早くて17.5回、一番遅くて27.5回のチェックが行なわれ、また、

4回をカウント (1.5 P) . . . . 早くて1.5 P, 遅くて7.5 P。

(ロ) 2枚目の用紙が、ある空きマスのチェックで印がついた5箇所の内の4箇所にチェックが  
入る場合：

周りの列とブロックには平均して7個の表示数がある。

一つの空きマスのチェックがすむ . . . . 7 P,

5箇所のチェックを確認 . . . . 早くて4 P, 遅くて5 P。

・該当数があるかないかの割合を  $1/5, 4/5$  とし、その  $1/5$  の内の  $5/6$  が (イ) 型であるとする。

該当数がない場合のポイントは、 $4+5+5+27.5+7.5+5 \times (7+5) = 109 P$

(イ) 型の該当数がある場合のポイントは、 $4+5+22.5+4.5 = 36 P$

(ロ) 型の該当数がある場合のポイントは、 $14+35+5 \times (7+5)/2 = 79 P$ 。

以上から、求めるポイントは、

$$1/5 \times 5/6 \times 36 + 1/5 \times 1/6 \times 79 + 4/5 \times 109 = 95.8 P。$$

(7) 6-空きマス技法のポイント

同様にして、127 P となる。

さて、最後に、上に得た8個のポイントの簡単な比を求める。

$$8 : 29 : 20 : 16.5 : 46.3 : 69 : 95.8 : 127$$

$$\simeq 2 : 7.5 : 5 : 4 : 11.5 : 17 : 24 : 32$$

#### 4.3 172題の評価値のまとめ

##### 1) レベル1

- ・問題数：13
- ・平均値：1.458
- ・標準偏差：0.332
- ・度数分布表：

以上	未満	度数
1.0	～ 1.2	3
1.2	～ 1.4	2
1.4	～ 1.6	3
1.6	～ 1.8	3
1.8	～	2

##### 2) レベル2

- ・問題数：75
- ・平均値：2.201
- ・標準偏差：0.488
- ・度数分布表：

以上	未満	度数
1.3	～ 1.6	10
1.6	～ 1.9	11
1.9	～ 2.2	16
2.2	～ 2.5	17
2.5	～ 2.8	13
2.8	～ 3.1	3
3.1	～ 3.4	5

##### 3) レベル3

- ・問題数：84
- ・平均値：3.003
- ・標準偏差：0.601
- ・度数分布表：

以上	未満	度数
1.6	～ 2.0	3
2.0	～ 2.4	11
2.4	～ 2.8	16
2.8	～ 3.2	27
3.2	～ 3.6	13
3.6	～ 4.0	11
4.0	～	3

(注意4.1) 評価値を計算するコンピュータプログラムは、6節でより一般的なものを示すので、ここでは省略する。また、この方法では解けない[2]の問題88番については、その評価値としてレベル3の最高評価値4.759をとった。

#### 4.4 難易度の基準

[基準B] その評価値と解法に必要な  $n$  - 空きマス技法の  $n$  の値により、

- 1) 評価値が1.75未満ならレベル1,
- 2) 評価値が1.75以上2.6未満ならレベル2,
- 3)  $n \geq 5$ か評価値が2.6以上ならレベル3とする。

・この基準の決め方について：

① 評価値がそれ以上かまたはそれ以下で除外されるものが各レベルの問題数の20%以内にそれぞれおさまる。

② 各レベル間の仕切りの値を、それぞれの平均値の間隔を標準偏差による比例配分で決めた値を目安にし、切りのよい数値とする。

例えば、レベル1とレベル2の場合、

$$1.458 + (2.201 - 1.458) \times 0.332 / (0.332 + 0.488) \simeq 1.759$$

また、レベル2とレベル3の場合、

$$2.201 + (3.003 - 2.201) \times 0.488 / (0.488 + 0.601) \approx 2.56$$

・この基準に従うと、

- ① レベル1の内、評価値が1.75以上のものは2題で、20%以内に収まる。
- ② レベル2の内、評価値が1.75未満のものは14題で、20%以内に収まる。
- ③ レベル2の内、評価値が2.6以上のものは15題で、丁度20%となる。
- ④ レベル3の内、評価値が2.6未満のもの（但し $n \geq 5$ のものは除く）は19題で、23%以内に収まる。

・解決に必要な $n$ -空きマス技法の $n$ と基準Bによるレベルとの間の関係

	1-空きマス 技法まで	2-空きマス 技法まで	3-空きマス 技法まで	4-空きマス 技法まで	5-空きマス 技法以上	計
レベル1	3	10	12	0	0	25
レベル2	0	4	34	29	0	67
レベル3	0	0	1	40	39	80
計	3	14	47	69	39	172

これから、求める基準が得られる。

[基準1] その評価値と解決に必要な $n$ -空きマス技法の $n$ の値により、

- 1)  $n \leq 2$  であるか、あるいは $n = 3$ かつ評価値が1.75未満ならレベル1、
- 2)  $n = 3$  で評価値が1.75以上2.6未満であるか、 $n = 4$  で評価値が1.75未満ならレベル2、
- 3)  $n \geq 5$  であるか、あるいは $n = 4$  かつ評価値が2.6以上ならレベル3とする。

(注意4.2) レベル2の基準として $n \geq 3$ としたのは、67題の中で $n < 3$ となるのは4題、すなわち6%以内ということ、及びレベル2の難易度として相応しいから。

(注意4.3) 基準Iを分かりやすく言えばつぎのようになる：

① レベル3は、解決に必要な空きマス技法の値が5であるか、あるいは4であって、かつ4ないし3-空きマス技法が約5回ぐらい必要なやっかいな手間のかかる問題。

② レベル1は、解決に必要な空きマス技法の値が3以内であって、2ないし3-空きマス技法が約2,3回ですむそんなに手間のかからない問題。

③ レベル2はその中間。解決に必要な空きマス技法の値が3以上であって、かつ4ないし3-空きマス技法が約3,4回ぐらい必要なやや手間のかかる問題。

## 5. [4],[5]の264題の結果

5.1 [4],[5]では問題の難易度がレベル1からレベル5まで5段階あり、4節にあげた解法による264題の結果はつぎのようにまとめられる。

1) レベル1	2) レベル2	3) レベル3
・問題数：58	・問題数：58	・問題数：50
・平均値：0.956	・平均値：1.673	・平均値：2.153
・標準偏差：0.363	・標準偏差：0.487	・標準偏差：0.402
・未解決数：0	・未解決数：0	・未解決数：3題(6%)
4) レベル4	5) レベル5	
・問題数：48	・問題数：46	
・平均値：2.658	・平均値：2.832	
・標準偏差：0.384	・標準偏差：0.329	
・未解決数：16題(33.3%)	・未解決数：25題(54.3%)	

### 5.2 この結果からの考察

1) レベル1とレベル2の間は平均値の差が0.72あり、またレベル2とレベル3の間、レベル3とレベル4の間はそれぞれ0.48,0.5あるが、レベル4とレベル5の間は0.17と差が極端に小さい。

2) 4節であげた解法では解けない問題がレベル4とレベル5には多く含まれている。

以上の2点から、新しい評価基準の必要性が問われていると考えられる。

そこで、新しい評価基準を導入しレベル3からレベル5までの問題につき検討する。

## 6. 難易度の基準化(3)

### 6.1 グループ確定数を用いた技法

(定義) あるタテ列かヨコ列、あるいはあるブロックの表字数の残りから決まるグループ確定数をタイプ1のグループ確定数と呼ぶ。また、2ペア数は核ブロック数になるが、ある核ブロック空き列において、その空きマスと同数の2ペア数がある核ブロック数となればそれらはグループ確定数となり、これを2ペアグループ確定数とよぶ。

さて、あるグループ確定数において、仮の順序でそれら数字を表示数と見なす。対応する核ブロック空き列を一部として含むタテ列かヨコ列、あるいはブロックを考え、その空きマス群をとるとき、この空きマス群を補-空きマス群と呼ぶ。特に、この空きマス群があるブロックに集中し、核ブロック空き列となると、補-核ブロック空き列と呼び、また、この核ブロッ

ク空き列に対応するグループ確定数を補-グループ確定数と呼ぶ。

[技法4] グループ確定数による補-空きマス群に対し、技法3を用いる。

[技法5] グループ確定数において、仮の順序でそれら数字を表示数と見なし、技法2を用いる。

[技法6] グループ確定数に対し、対応するその核ブロック空き列に直交するタテないしヨコの3つのマス列の組（技法2の候補となる列の組）をとり、それぞれのマス列の組（3組ある）に技法2を用いる。

（実例）右の図3を考える。

① ヨコ6列から決まるタイプ1のグループ確定数{8, 9)のブロック4の補-空きマス群{3, 6, 7)に技法3-すなわち、技法4-を用いると、数6が決まる（図の○で囲ったもの）。② ヨコ6列から決まるタイプ1のグループ確定数{8, 9)に対し、技法5を用いて数8が決まる（図の○で囲ったもの）。

③ ブロック1で決まるタイプ1のグループ確定数確定数{3, 5, 6)のヨコ3列の補-グループ確定数{2, 7, 8)に技法5を適用すると、数7が決まる（図の○で囲ったもの）。④ ブロック8において、{1, 4)は2ペア

(図3)

2	7	8		1	4			
9	4	1				⑦		8
			⑦			1	4	9
1	2	4	3			⑧	9	
	⑥		8					
		5	2	4	1	6	7	3
	3	9	5				6	
			*			9		
6			*	⑨	3			

グループ確定数である（その核ブロック空き列は\*印をつけたもの）。タテ4列におけるこの補-空きマス群{6, 7, 9)に技法3-すなわち、技法4を用いると、数7が決まる（図の□で囲ったもの）。③で決まった表示数7を用いた）。⑤ ブロック8の2ペア確定数{1, 4)に技法6を用いる、すなわち、ヨコ7列, 8列, 9列の組をとり、技法2を適用すると、数9が決まる（図の○で囲ったもの）。

## 6.2 技法7の導入

[技法7] 2つの数字（数字対）が入る可能性をもつ空きマスを選び、その一方の数字を仮に入れて解を求める処理を続ける。このとき、

- ① 完成すれば、それが求める解である、
- ② 途中で矛盾が出れば、仮の数字の対である他方の数字がこのマスの表示数として決まる、
- ③ 矛盾が出ないまま継続作業が途中でストップしたら、対で可能な新しい空きマスに代えて、上記処理を新規に行なう。

（注意6.1）技法7は2節で述べた処理3に相当する。

さて、これらの技法の導入による[4], [5]のレベル3以上の148題の解法結果を示す。



	3-空きマス 技法まで	4-空きマス 技法まで	5-空きマス 技法以上	技法4 が必要	技法5 が必要	技法6 が必要	技法7 使用	計
レベル3	26	18	3	0	0	0	3	50
レベル4	7	18	11	4	2	0	10	52
レベル5	0	12	9	9	0	1	15	46
計	33	48	23	13	2	1	28	148

この結果から、つぎのことが考えられる。

1. 1) 4-空きマス技法までで解ける範囲をレベル3とする。
- 2) レベル4、5とも、4-空きマス技法以上のものとする。

を、この場合の基準とするとよい。

2. 特に、5-空きマス技法以上+技法4~7必要の割合について、

レベル3：12%、レベル4：52%、レベル5：74%

より、テクニック（難しいテクニック）が重要な要素であることがわかる。

### 6.3 基準化のための評価値について

基本的に4.1と同様なので、違っているところだけあげておく。

- (1) 評価値は、用いた技法による平均ポイント数+テクニカルポイント数(解法に必要な技法4~6および技法7の使用に応じたもの)で決まる。
- (2) 使用する技法は、今までにあげた技法1から技法7までの7通りとする。
- (3イ) 技法に応じたポイントをつぎのようにとる。

- |   |                 |      |      |      |
|---|-----------------|------|------|------|
| ① | はじめの確定数         | .... | 0    | ポイント |
| ② | 技法1によって確定する     | .... | 2    | "    |
| ③ | 技法2 " "         | .... | 4    | "    |
| ④ | 技法3で2-空きマス技法による | .... | 4    | "    |
| ⑤ | " 3- " "        | .... | 11.5 | "    |
| ⑥ | " 4- " "        | .... | 17   | "    |
| ⑦ | " 5- " "        | .... | 24   | "    |
| ⑧ | 技法4によって確定する     |      |      |      |

グループ確定数が決まるのに要したポイント数（但し、複数のグループ確定数に関わる場合、それらの要したポイント数の総和をとる）に、用いた  $n$ -空きマス技法のポイント数を加える。

- ⑨ 技法5によって確定する

グループ確定数が決まるのに要したポイント数（但し、複数のグループ確定数に関わる場合、その中の最大ポイント数をとる）に、技法2のポイント数4を加える。

⑩ 技法6によって確定する

グループ確定数が決まるのに要したポイント数に、ポイント数5を加える。

⑪ 技法7の使用に関して

数字対の一方（または矛盾による他方）として確定 …… 4ポイント

(3ロ) グループ確定数が決まるのに要したポイント数

① タイプ1のグループ確定数 …… 2

② 2ペアグループ確定数 …… 3

③ 補グループ確定数 …… 元になったグループ確定数のポイント数+1

(3ハ) テクニカルポイント数

技法4が必要 …… 0.35, 技法5まで必要 …… 0.5, 技法6まで必要 …… 0.65

技法7の使用 …… 0.8

(4) 5-空きマス技法まで使用する。空きマスが6以上の場合は処理を技法7に移す。

(注意6.2) 4節でのポイント数設定は“手間”を重視したのに対し、6.2で指摘したように、ここでは“テクニック”を重視している。

① 技法2の“テクニック”ポイントを4とした（4節では5であったが）。

② 技法4のタイプ1のグループ確定数による1-空きマス技法はただの1-空きマス技法より6.5（ポイント数でいえば1.625）だけ手間が増えている。しかし、この技法の重要さはそのアイデア（テクニック）にある。従って、技法4ではただの $n$ -空きマス技法の場合より一律にグループ確定数を決めるのに要したポイント数だけ増やした。

技法5,6についても同様に行なった。

③ 技法4のテクニカルポイントの決め方：

[4], [5]の4節の解法で解けたレベル5の21題の内、14題が技法4の使用によりその評価値が減少する。その減少した値の平均値は0.329なので、切りの良い数値0.35をそのテクニカルポイント数とした。

同様に、技法7の使用のポイント数については、[4], [5]の4節の解法で解けた6-空きマス技法の3題について、その減少した値の平均値をとると0.759なので、切りの良い0.8をテクニカルポイント数とした。

さらに、技法5、技法6のテクニカルポイント数は、この技法4と技法7の比例配分によって決めた。

6.4 上に述べた解法のコンピュータプログラムを示す。

ただし、優先順位をヨコ列→タテ列→ブロックとしたものに限定している。

このプログラムの特徴は、補-グループ確定数を求める時や技法4~6の使用時で用いる「仮の順序でそれら数字を表示数と見なす」という処理部である（プログラムの中で仮置き・もどしとコメントがついた部分）。

```

1000 REM *** Number Place ***
1010 REM == nanido_kijyun(2) ==
1020 DEF FNBL(I)=INT((I - 1)/3)+1
1030 '
1040 DIM M(9,9),HM(9,9),A(9),AA(9,9),GS(9),N(9),CHS(9)
1050 DIM GTY(60),GI(60),GJ(60),GSU(60),AP(3),SG(4),P(9),PI(9,5),PJ(9,5)
1060 DIM GRTY(60),GRI(60),GRJ(60),GRD(60),GRP(60),GRK(60),GRIT(60,3),GRSU(60,3)
1070 DIM BR(15),HBR(15),PT(25),GRKP(6),HGRKP(6),HGN(60),ITI(2),TSU(9),KOSU(2)
1080 CONSOLE „0,1: SCREEN 3: CLS 3
1090 ' --- syoki_settei ---
1100 KAKUTEI=0
1110 FOR I=1 TO 9:FOR J=1 TO 9: M(I,J)=0: NEXT J,I
1120 FOR I=1 TO 15: BR(I)=0: NEXT I
1130 FOR I=1 TO 6: GRKP(I)=0: NEXT I
1140 ' --- data_yomi
1150 ' & syoki_jyoho - input ---
1160 FILE$="a:nprob.dat"
1170 OPEN FILE$ FOR INPUT AS #1
1180 WHILE NOT EOF(1)
1190 INPUT #1,TA,YO,SU
1200 GOSUB *KAKUTEI
1210 WEND
1220 ' --- su_kakutei ---
1230 GOSUB *FIRSTGIHOU2
1240 FEND=0: FGIHOU7=0: FMUJYUN=0
1250 WHILE FEND=0
1260 FEGIHOU16=0
1270 WHILE FEGIHOU16=0
1280 TUIKA=0
1290 FOR NAKI=1 TO 5

```

```
1300 IF NAKI=1 THEN
1310 ' -- gihou_1 --
1320 GOSUB *GIHOU3
1330 IF (TUIKA=1)OR(FMUJYUN=1) THEN GOTO *OUTLOOP
1340 ELSE
1350 ' -- gihou_2 --
1360 IF NAKI=2 THEN
1370 FH1=0:GOSUB *GIHOU2
1380 IF TUIKA=1 THEN GOTO *OUTLOOP
1390 END IF
1400 ' -- gihou_3 --
1410 GOSUB *GIHOU3
1420 IF (TUIKA=1)OR(FMUJYUN=1) THEN GOTO *OUTLOOP
1430 ' -- group_kaku sakusei --
1440 GOSUB *MAKEGROUP
1450 ' -- naki=1.gihou_4 --
1460 IF NAKI=2 THEN
1470 NAKI=1 ' kari_oki
1480 GOSUB *GIHOU41
1490 IF (TUIKA=1)OR(FMUJYUN=1) THEN GOTO *OUTLOOP
1500 GOSUB *GIHOU42
1510 IF (TUIKA=1)OR(FMUJYUN=1) THEN GOTO *OUTLOOP
1520 NAKI=2 ' modosi
1530 END IF
1540 ' -- gihou_4 --
1550 GOSUB *GIHOU41
1560 IF (TUIKA=1)OR(FMUJYUN=1) THEN GOTO *OUTLOOP
1570 GOSUB *GIHOU42
1580 IF (TUIKA=1)OR(FMUJYUN=1) THEN GOTO *OUTLOOP
1590 ' -- gihou_5,6 --
1600 IF NAKI=2 THEN
1610 GOSUB *GIHOU5
1620 IF TUIKA=1 THEN GOTO *OUTLOOP
1630 GOSUB *GIHOU6
1640 IF TUIKA=1 THEN GOTO *OUTLOOP
1650 END IF
1660 END IF
```

```
1670 NEXT NAKI
1680 *OUTLOOP
1690 IF (TUIKA=0)OR(FMUJYUN=1) THEN FEGIHOU16=1
1700 WEND
1710 IF KAKUTEI=81 THEN GOTO *KAIEND
1720 ' -- gihou_7 --
1730 GIHOU=7: BUNRN=15
1740 IF FGIHOU7=0 THEN
1750 FGIHOU7=1:FMUJYUN=0
1760 ' hikae_toru
1770 HKAKUTEI=KAKUTEI
1780 FOR I=1 TO 9:FOR J=1 TO 9: HM(I,J)=M(I,J): NEXT J,I
1790 FOR I=1 TO 15: HBR(I)=BR(I): NEXT I
1800 FOR I=1 TO 6: HGRKP(I)=GRKP(I): NEXT I
1810 ' koho_sentak
1820 KOTA=1:KOYO=0: GOSUB *KOHOSSEN
1830 ELSE
1840 ' atai_modosi
1850 KAKUTEI=HKAKUTEI
1860 FOR I=1 TO 9:FOR J=1 TO 9: M(I,J)=HM(I,J): NEXT J,I
1870 FOR I=1 TO 15: BR(I)=HBR(I): NEXT I
1880 FOR I=1 TO 6: GRKP(I)=HGRKP(I): NEXT I
1890 ' mujyun_ari?
1900 IF FMUJYUN=0 THEN
1910 GOSUB *KOHOSSEN
1920 ELSE
1930 FGIHOU7=0:FMUJYUN=0
1940 ' kakutei_syori
1950 TA=KOTA:YO=KOYO:SU=KOSU(2): GOSUB *KAKUTEI
1960 END IF
1970 END IF
1980 WEND
1990 *KAIEND
2000 ' --- kai_hyouji ---
2010 GOSUB *PRIN
2020 GOSUB *POINTMEAN
2030 END
```

```
2040 ::::::::::
2300 *PRIN
2310 XGS=168:XGL=384
2320 YGS=40:YGL=328
2330 ' -- waku_hyouji --
2340 FOR YL=YGS TO YGL STEP 32
2350 LINE (XGS,YL) - (XGL,YL)
2360 IF (YL - YGS) MOD 96=0 THEN LINE (XGS,YL+1) - (XGL,YL+1)
2370 NEXT YL
2380 FOR XL=XGS TO XGL STEP 24
2390 LINE (XL,YGS) - (XL,YGL)
2400 IF (XL - XGS) MOD 72=0 THEN LINE (XL+1,YGS) - (XL+1,YGL)
2410 NEXT XL
2420 ' -- suu_hyouji --
2430 FOR TA=1 TO 9:FOR YO=1 TO 9
2440 XHYJ=21+3*(YO - 1): YHYJ=3+2*(TA - 1)
2450 LOCATE XHYJ,YHYJ: PRINT M(TA,YO);
2460 NEXT YO,TA
2470 RETURN
2480 :::
2500 *KAKUTEI
2510 ' == par. TA,YO,SU,BUNRN ==
2520 KAKUTEI=KAKUTEI+1: TUIKA=1
2530 M(TA,YO)=SU
2540 '
2550 BR(BUNRN)=BR(BUNRN)+1
2560 IF (4<=GIHOU)AND(GIHOU<=6) THEN GRKP(GRKP)=GRKP(GRKP)+1
2570 RETURN
2580 :::
2600 *FIRSTGIHOU2
2610 REP=1
2620 WHILE REP=1
2630 REP=0
2640 BUNRN=0: TUIKA=0
2650 FOR TY=1 TO 2
2660 ' ty=1 -- yoko ty=2 -- tate
2670 FOR II=1 TO 3
```

```
2680   CHN=9
2690   FOR Q9=1 TO CHN: CHS(Q9)=Q9: NEXT Q9
2700   FH1=0:GOSUB *TECH2
2710   IF TUIKA=1 THEN REP=1
2720   NEXT II
2730   NEXT TY
2740   WEND
2750   RETURN
2760   :::::
2800 *HOSIKI2
2810 ' == par. TA,YO ==
2820 GOSUB *SIYOUSUU
2830 IF PN=8 THEN SU=KN:GOSUB *KAKUTEI
2840 RETURN
2850 :::
2900 *SIYOUSUU
2910 FOR U=1 TO 9: N(U)=0: NEXT U
2920 '
2930 FOR U=1 TO 9: N(M(TA,U))=1: NEXT U
2940 FOR U=1 TO 9: N(M(U,YO))=1: NEXT U
2950 B1=FNBL(TA):B2=FNBL(YO)
2960 FOR U1=3*B1 - 2 TO 3*B1:FOR U2=3*B2 - 2 TO 3*B2
2970   N(M(U1,U2))=1
2980 NEXT U2,U1
2990 '
3000 PN=0:SYS=0
3010 FOR U=1 TO 9
3020   IF N(U)=1 THEN PN=PN+1 ELSE KN=U:SYS=SYS+1:TSU(SYS)=U
3030 NEXT U
3040 RETURN
3050 ::::
3100 *BLOCKSU
3110 REP=1
3120 WHILE REP=1
3130   REP=0: H4=0
3140   FH1=1:GOSUB *GIHOU2
3150   IF TUIKA=1 THEN REP=1
```

```
3160 WEND
3170 RETURN
3180 :::::
3200 *GIHOU2
3210 ' == par. FH1 ==
3220 GIHOU=2:BUNRN=1
3230 FOR TY=1 TO 2
3240 ' ty=1 -- yoko ty=2 -- tate
3250 FOR II=1 TO 3
3260   CHN=9
3270   FOR Q9=1 TO CHN: CHS(Q9)=Q9: NEXT Q9
3280   GOSUB *TECH2
3290   IF (FH1=0)AND(TUIKA=1) THEN RETURN
3300   NEXT II
3310 NEXT TY
3320 RETURN
3330 ::
3400 *TECH2
3410 ' == par. FH1,TY,II, CHN,CHS() ==
3420 ' syoki_ka
3430 FOR I9=1 TO 9
3440   P(I9)=0
3450   FOR J9=0 TO 3: PI(I9,J9)=0:PJ(I9,J9)=0: NEXT J9
3460   NEXT I9
3470   '
3480   FOR I9=3*II - 2 TO 3*II
3490     FOR J9=1 TO 9
3500       IF TY=1 THEN MS=M(I9,J9) ELSE MS=M(J9,I9)
3510       IF MS>0 THEN
3520         P(MS)=P(MS)+1
3530         PI(MS,P(MS))=I9:PJ(MS,P(MS))=FNBL(J9)
3540       END IF
3550     NEXT J9
3560   NEXT I9
3570   '
3580   FOR Z=1 TO CHN
3590     SU=CHS(Z)
```



```

3600 IF P(SU)=2 THEN
3610 GRI=9*(II - 1)+6 - (PI(SU,1)+PI(SU,2)):GRJ=6 - (PJ(SU,1)+PJ(SU,2))
3620 PN=0
3630 FOR J9=3*GRJ - 2 TO 3*GRJ
3640 IF TY=1 THEN TA=GRI:YO=J9 ELSE TA=J9:YO=GRI
3650 IF M(TA,YO)>0 THEN
3660 PN=PN+1
3670 ELSE
3680 FOR J8=1 TO 9
3690 IF TY=1 THEN MS=M(J8,YO) ELSE MS=M(TA,J8)
3700 IF MS=SU THEN PN=PN+1:GOTO *EXJ8
3710 NEXT J8
3720 JN=J9
3730 *EXJ8
3740 END IF
3750 NEXT J9
3760 '
3770 IF PN=2 THEN
3780 IF TY=1 THEN TA=GRI:YO=JN ELSE TA=JN:YO=GRI
3790 IF FH1=0 THEN GOSUB *KAKUTEI: RETURN
3800 ELSE
3810 IF FH1=1 THEN
3820 H4=H4+1
3830 GTY(H4)=TY:GSU(H4)=SU:GI(H4)=GRI:GJ(H4)=GRJ
3840 END IF
3850 END IF
3860 END IF
3870 NEXT Z
3880 RETURN
3890 ::::
4000 *GIHOU3
4010 GIHOU=3:MBUNRN=1
4020 ' yoko
4030 TTY=1
4040 FOR I=1 TO 9
4050 GOSUB *TECH3
4060 IF TUIKA=1 THEN RETURN

```

```
4070 NEXT I
4080 'tate
4090 TTY=2
4100 FOR I=1 TO 9
4110 GOSUB *TECH3
4120 IF TUIKA=1 THEN RETURN
4130 NEXT I
4140 'block
4150 TTY=3
4160 FOR BI=1 TO 3:FOR BJ=1 TO 3
4170 GOSUB *TECH3
4180 IF TUIKA=1 THEN RETURN
4190 NEXT BJ,BI
4200 RETURN
4210 :::
4300 *TECH3
4310 ' == par. TTY,I(or BI,BJ) ==
4320 ' aki_check
4330 GOSUB *AKICH
4340 ' -- kakutei_syori --
4350 BUNRN=MBUNRN+AKI
4360 IF (1<=AKI)AND(AKI<=NAKI) THEN
4370 ' -- hosiki_1 --
4380 FOR S9=1 TO AKI
4390 SU=GS(S9)
4400 PN=0
4410 SELECT TTY
4420 CASE 1,2
4430 FOR J9=1 TO AKI
4440 IF TTY=1 THEN TA=I:YO=A(J9) ELSE TA=A(J9):YO=I
4450 GOSUB *ITTICH
4460 IF FITTI=1 THEN PN=PN+1 ELSE JN=A(J9)
4470 NEXT J9
4480 IF TTY=1 THEN TA=I:YO=JN ELSE TA=JN:YO=I
4490 CASE 3
4500 FOR K9=1 TO AKI
4510 TA=AA(K9,1):YO=AA(K9,2): GOSUB *ITTICH
```

```

4520     IF FITTI=1 THEN PN=PN+1 ELSE IN=AA(K9,1):JN=AA(K9,2)
4530     NEXT K9
4540     TA=IN:YO=JN
4550 END SELECT
4560 '
4570 IF PN=AKI - 1 THEN
4580   GOSUB *KAKUTEI
4590   RETURN
4600 END IF
4610 ' -- mujyun_check --
4620 IF (FGIHOU7=1)AND(PN=AKI) THEN FMUJYUN=1: RETURN
4630 NEXT S9
4640 ' -- hosiki_2 --
4650 FOR K9=1 TO AKI
4660   IF TTY=1 THEN TA=I:YO=A(K9)
4670   IF TTY=2 THEN TA=A(K9):YO=I
4680   IF TTY=3 THEN TA=AA(K9,1):YO=AA(K9,2)
4690   GOSUB *HOSIKI2
4700   IF TUIKA=1 THEN RETURN
4710 NEXT K9
4720 END IF
4730 RETURN
4740 ::::
4750 *AKICH
4760 ' == par. TTY ==
4770 FOR I9=1 TO 9: N(I9)=0: NEXT I9
4780 '
4790 AKI=0
4800 SELECT TTY
4810   CASE 1,2
4820     FOR J9=1 TO 9
4830       IF TTY=1 THEN MS=M(I,J9) ELSE MS=M(J9,I)
4840       IF MS=0 THEN AKI=AKI+1:A(AKI)=J9 ELSE N(MS)=1
4850     NEXT J9
4860   CASE 3
4870     FOR I9=3*BI - 2 TO 3*BI:FOR J9=3*BJ - 2 TO 3*BJ
4880       IF M(I9,J9)=0 THEN AKI=AKI+1:AA(AKI,1)=I9:AA(AKI,2)=J9 ELSE N(M(I9,J9))=1

```

```
4890 NEXT J9,I9
4900 END SELECT
4910 '
4920 GSN=0
4930 FOR I9=1 TO 9
4940 IF N(I9)=0 THEN GSN=GSN+1:GS(GSN)=I9
4950 NEXT I9
4960 RETURN
4970 :::
4980 *ITTICH
4990 ' == par. TA,YO,SU ==
5000 FITTI=0
5010 FOR U9=1 TO 9
5020 IF M(TA,U9)=SU THEN FITTI=1
5030 IF M(U9,YO)=SU THEN FITTI=1
5040 NEXT U9
5050 B1=INT((TA - 1)/3)+1:B2=INT((YO - 1)/3)+1
5060 FOR U9=3*B1 - 2 TO 3*B1:FOR U8=3*B2 - 2 TO 3*B2
5070 IF M(U9,U8)=SU THEN FITTI=1
5080 NEXT U8,U9
5090 RETURN
5100 :::
5110 *BLNARCH
5120 GSY=0
5130 IF TTY=3 THEN
5140 FOR GR=1 TO 2
5150 BEN=0: HB=AA(1,GR)
5160 FOR K9=2 TO AKI
5170 IF HB<>AA(K9,GR) THEN BEN=1
5180 NEXT K9
5190 IF BEN=0 THEN GSY=1:HGR=GR
5200 NEXT GR
5210 ELSE
5220 BEN=0: HB=FNBL(A(1))
5230 FOR K9=2 TO AKI
5240 IF HB<>FNBL(A(K9)) THEN BEN=1
5250 NEXT K9
```

```
5260 IF BEN=0 THEN GSY=1
5270 END IF
5280 RETURN
5290 :::
5300 *MAKEGROUP
5310 GKO=0
5320 GOSUB *LISTGRP1
5330 FH1=1:GOSUB *BLOCKSU
5340 IF H4>0 THEN GOSUB *LISTGRP2
5350 ' ho_group sakusei
5360 HOD=0:GOSUB *HOGROUP
5370 HOD=1:GOSUB *HOGROUP
5380 GOSUB *DBHGRP
5390 RETURN
5400 :::
5500 *LISTGRP1
5510 MKGRPN=1
5520 ' yoko(tty=1), tate(tty=2)
5530 FOR TTY=1 TO 2
5540 FOR I=1 TO 9
5550 GOSUB *MKGRP
5560 NEXT I
5570 NEXT TTY
5580 ' block
5590 FOR BI=1 TO 3:FOR BJ=1 TO 3
5600 TTY=3: GOSUB *MKGRP
5610 NEXT BJ,BI
5620 RETURN
5630 :::
5700 *LISTGRP2
5710 FOR W=1 TO H4 - 1
5720 TTY=GTY(W):GI=GI(W):GJ=GJ(W):SU=GSU(W)
5730 ' kanou_su
5740 CN=1: SG(1)=SU
5750 FOR J9=W+1 TO H4
5760 IF (TTY=GTY(J9))AND(GI=GI(J9))AND(GJ=GJ(J9)) THEN
5770 CN=CN+1:SG(CN)=GSU(J9)
```

```
5780 END IF
5790 NEXT J9
5800 '
5810 IF CN>1 THEN
5820 ' jyogai_su
5830 JGSU=0
5840 FOR K9=1 TO 3
5850 GJK=3*(GJ - 1)+K9
5860 IF TTY=1 THEN MS=M(GI,GJK) ELSE MS=M(GJK,GI)
5870 IF MS>0 THEN
5880 JGSU=JGSU+1:JG(JGSU)=K9
5890 ELSE
5900 CIT=0
5910 FOR KP=1 TO CN
5920 FOR Q8=1 TO 9
5930 IF TTY=1 THEN MS=M(Q8,GJK) ELSE MS=M(GJK,Q8)
5940 IF MS=SG(KP) THEN CIT=CIT+1
5950 NEXT Q8
5960 NEXT KP
5970 IF CIT=CN THEN JGSU=JGSU+1:JG(JGSU)=K9
5980 END IF
5990 NEXT K9
6000 IF CN=3 - JGSU THEN
6010 FOR K8=1 TO 3: N(K8)=1: NEXT K8
6020 FOR K8=1 TO JGSU: N(JG(K8))=0: NEXT K8
6030 APN=0
6040 FOR K8=1 TO 3
6050 IF N(K8)=1 THEN APN=APN+1:AP(APN)=3*(GJ - 1)+K8
6060 NEXT K8
6070 ' kiroku
6080 GKO=GKO+1
6090 GRD(GKO)=0:GRP(GKO)=3
6100 GRTY(GKO)=TTY:GRI(GKO)=GI:GRJ(GKO)=GJ
6110 GRK(GKO)=CN
6120 FOR K9=1 TO CN
6130 GRIT(GKO,K9)=AP(K9):GRSU(GKO,K9)=SG(K9)
6140 NEXT K9
```

```
6150 END IF
6160 END IF
6170 NEXT W
6180 RETURN
6190 :::
6200 *HOGROUP
6210 MKGRPN=2:SGKO=GKO
6220 FOR W=1 TO SGKO
6230 IF GRD(W)=HOD THEN
6240 TTY=GRTY(W):GI=GRI(W):GJ=GRJ(W):GD=GRD(W):GP=GRP(W)
6250 ' kari_oki
6260 FOR K9=1 TO GRK(W)
6270 IF TTY=1 THEN TA=GI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=GI
6280 M(TA,YO)=GRSU(W,K9)
6290 NEXT K9
6300 KTY=TTY
6310 ' retu, block
6320 I=GI:GOSUB *MKGRP
6330 TTY=3:BI=FNBL(GI):BJ=GJ
6340 GOSUB *MKGRP
6350 ' M_modosi
6360 FOR K9=1 TO GRK(W)
6370 IF KTY=1 THEN TA=GI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9): YO=GI
6380 M(TA,YO)=0
6390 NEXT K9
6400 END IF
6410 NEXT W
6420 RETURN
6430 :::::
6500 *DBHOGRP
6510 MKGRPN=3:SGKO=GKO
6520 FOR W=1 TO SGKO - 1
6530 DTY=GRTY(W):DI=GRI(W):DJ=GRJ(W):DP=GRP(W)
6540 ' kari_oki
6550 FOR K9=1 TO GRK(W)
6560 IF DTY=1 THEN TA=DI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=DI
6570 M(TA,YO)=GRSU(W,K9)
```

```
6580 NEXT K9
6590 KDTY=DTY
6600 ' retu
6610 FOR X=W+1 TO SGKO
6620 IF (DTY=GRTY(X))AND(DI=GRI(X)) THEN
6630 TTY=GRTY(X):GI=GRI(X):GJ=GRJ(X):GP=GRP(X)
6640 ' kari_oki
6650 FOR K9=1 TO GRK(X)
6660 IF TTY=1 THEN TA=GI:YO=GRIT(X,K9) ELSE TA=GRIT(X,K9): YO=GI
6670 M(TA,YO)=GRSU(X,K9)
6680 NEXT K9
6690 '
6700 I=GI:GOSUB *MKGRP
6710 ' M_modosi
6720 FOR K9=1 TO GRK(X)
6730 IF TTY=1 THEN TA=GI:YO=GRIT(X,K9) ELSE TA=GRIT(X,K9): YO=GI
6740 M(TA,YO)=0
6750 NEXT K9
6760 END IF
6770 NEXT X
6780 ' block
6790 IF DTY=1 THEN DBI=FNBL(DI):DBJ=DJ ELSE DBI=DJ:DBJ=FNBL(DI)
6800 FOR X=W+1 TO SGKO
6810 TTY=GRTY(X):GI=GRI(X):GJ=GRJ(X)
6820 IF TTY=1 THEN BI=FNBL(GI):BJ=GJ ELSE BI=GJ:BJ=FNBL(GI)
6830 IF (DBI=BI)AND(DBJ=BJ) THEN
6840 ' kari_oki
6850 FOR K9=1 TO GRK(X)
6860 IF TTY=1 THEN TA=GI:YO=GRIT(X,K9) ELSE TA=GRIT(X,K9): YO=GI
6870 M(TA,YO)=GRSU(X,K9)
6880 NEXT K9
6890 KTY=TTY
6900 '
6910 TTY=3:GOSUB *MKGRP
6920 ' M_modosi
6930 FOR K9=1 TO GRK(X)
6940 IF KTY=1 THEN TA=GI:YO=GRIT(X,K9) ELSE TA=GRIT(X,K9): YO=GI
```



```
6950     M(TA,YO)=0
6960     NEXT K9
6970     END IF
6980     NEXT X
6990     ' M_modosi
7000     FOR K9=1 TO GRK(W)
7010     IF KDTY=1 THEN TA=DI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=DI
7020     M(TA,YO)=0
7030     NEXT K9
7040     NEXT W
7050     RETURN
7060     ::::::
7100     *MKGRP
7110     ' aki_check
7120     GOSUB *AKICH
7130     IF (AKI=2)OR(AKI=3) THEN
7140     ' -- block^narabi_check
7150     GOSUB *BLNARCH
7160     IF GSY=1 THEN
7170     ' kiroku_zumi?
7180     IF TTY<3 THEN QTY=TTY:QI=I:QJ=HB ELSE QTY=HGR:QI=AA(1,HGR):QJ=FNBL(AA(1,3 - HGR))
7190     FOR Z=1 TO GKO
7200     IF (QTY=GRTY(Z))AND(QI=GRI(Z))AND(QJ=GRJ(Z)) THEN RETURN
7210     NEXT Z
7220     ' kiroku
7230     GKO=GKO+1
7240     IF MKGRPN=1 THEN GRD(GKO)=0:GRP(GKO)=2
7250     IF MKGRPN=2 THEN GRD(GKO)=GD+1:GRP(GKO)=GP+1
7260     IF MKGRPN=3 THEN
7270     GRD(GKO)=3
7280     IF DP>GP THEN GRP(GKO)=DP+1 ELSE GRP(GKO)=GP+1
7290     END IF
7300     GRTY(GKO)=QTY:GRI(GKO)=QI:GRJ(GKO)=QJ
7310     GRK(GKO)=AKI
7320     FOR K9=1 TO AKI
7330     IF TTY<3 THEN GRIT(GKO,K9)=A(K9) ELSE GRIT(GKO,K9)=AA(K9,3 - HGR)
7340     GRSU(GKO,K9)=GS(K9)
```

```
7350 NEXT K9
7360 END IF
7370 END IF
7380 RETURN
7390 :::
7400 *GIHOU41
7410 GIHOU=4:MBUNRN=7
7420 FOR W=1 TO GKO
7430 TTY=GRTY(W):GI=GRI(W):GJ=GRJ(W)
7440 ' kari_oki
7450 FOR K9=1 TO GRK(W)
7460 IF TTY=1 THEN TA=GI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=GI
7470 M(TA,YO)=GRSU(W,K9)
7480 NEXT K9
7490 KTY=TTY
7500 ' retu
7510 GRKP=GRP(W)
7520 I=GI:GOSUB *TECH3
7530 IF TUIKA=0 THEN
7540 ' block
7550 TTY=3:BI=FNBL(GI):BJ=GJ:GOSUB *TECH3
7560 END IF
7570 ' M_modosi
7580 FOR K9=1 TO GRK(W)
7590 IF KTY=1 THEN TA=GI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=GI
7600 M(TA,YO)=0
7610 NEXT K9
7620 IF TUIKA=1 THEN RETURN
7630 NEXT W
7640 RETURN
7650 :::::
7700 *GIHOU42
7710 GIHOU=4:MBUNRN=7
7720 FOR W=1 TO GKO - 1
7730 DTY=GRTY(W):DI=GRI(W):DJ=GRJ(W)
7740 ' kari_oki
7750 FOR K9=1 TO GRK(W)
```

```
7760 IF DTY=1 THEN TA=DI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=DI
7770 M(TA,YO)=GRSU(W,K9)
7780 NEXT K9
7790 KDTY=DTY
7800 ' retu
7810 FOR X=W+1 TO GKO
7820 IF (DTY=GRTY(X))AND(DI=GRI(X)) THEN
7830 TTY=GRTY(X):GI=GRI(X):GJ=GRJ(X)
7840 ' kari_oki
7850 FOR K9=1 TO GRK(X)
7860 IF TTY=1 THEN TA=GI:YO=GRIT(X,K9) ELSE TA=GRIT(X,K9):YO=GI
7870 M(TA,YO)=GRSU(X,K9)
7880 NEXT K9
7890 '
7900 GRKP=GRP(W)+GRP(X)
7910 I=GI:GOSUB *TECH3
7920 ' M_modosi
7930 FOR K9=1 TO GRK(X)
7940 IF TTY=1 THEN TA=GI:YO=GRIT(X,K9) ELSE TA=GRIT(X,K9):YO=GI
7950 M(TA,YO)=0
7960 NEXT K9
7970 IF TUIKA=1 THEN GOTO *ESYDB
7980 END IF
7990 NEXT X
8000 ' block
8010 IF DTY=1 THEN DBI=FNBL(DI):DBJ=DJ ELSE DBI=DJ:DBJ=FNBL(DI)
8020 FOR X=W+1 TO GKO
8030 TTY=GRTY(X):GI=GRI(X):GJ=GRJ(X)
8040 IF TTY=1 THEN BI=FNBL(GI):BJ=GJ ELSE BI=GJ:BJ=FNBL(GI)
8050 IF (DBI=BI)AND(DBJ=BJ) THEN
8060 ' kari_oki
8070 FOR K9=1 TO GRK(X)
8080 IF TTY=1 THEN TA=GI:YO=GRIT(X,K9) ELSE TA=GRIT(X,K9):YO=GI
8090 M(TA,YO)=GRSU(X,K9)
8100 NEXT K9
8110 KTY=TTY
8120 '

```

```
8130 GRKP=GRP(W)+GRP(X)
8140 TTY=3:GOSUB *TECH3
8150 ' M_modosi
8160 FOR K9=1 TO GRK(X)
8170 IF KTY=1 THEN TA=GI:YO=GRIT(X,K9) ELSE TA=GRIT(X,K9):YO=GI
8180 M(TA,YO)=0
8190 NEXT K9
8200 IF TUIKA=1 THEN GOTO *ESYDB
8210 END IF
8220 NEXT X
8230 *ESYDB
8240 ' M_modosi
8250 FOR K9=1 TO GRK(W)
8260 IF KDTY=1 THEN TA=DI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=DI
8270 M(TA,YO)=0
8280 NEXT K9
8290 IF TUIKA=1 THEN RETURN
8300 NEXT W
8310 RETURN
8320 ::::::
8400 *GIHOU5
8410 GIHOU=5:BUNRN=13
8420 FOR W=1 TO GKO
8430 TTY=GRTY(W):II=FNBL(GRI(W))
8440 GK5=1:HGN(GK5)=W
8450 FOR X=W+1 TO GKO
8460 UTY=GRTY(X):UI=FNBL(GRI(X))
8470 IF (UTY=TTY)AND(UI=II) THEN GK5=GK5+1:HGN(GK5)=X
8480 NEXT X
8490 ' kari_oki
8500 FOR K9=1 TO GK5
8510 Y=HGN(K9)
8520 FOR K8=1 TO GRK(Y)
8530 IF GRTY(Y)=1 THEN TA=GRI(Y):YO=GRIT(Y,K8) ELSE TA=GRIT(Y,K8):YO=GRI(Y)
8540 M(TA,YO)=GRSU(Y,K8)
8550 NEXT K8
8560 NEXT K9
```

```

8570 ' siyou_su & grkp_kettei
8580 FOR K9=1 TO 9: N(K9)=0: NEXT K9
8590 GRKP=0
8600 FOR K9=1 TO GK5
8610 Y=HGN(K9)
8620 FOR K8=1 TO GRK(Y)
8630 N(GRSU(Y,K8))=1
8640 NEXT K8
8650 IF GRKP<GRP(Y) THEN GRKP=GRP(Y)
8660 NEXT K9
8670 CHN=0
8680 FOR K9=1 TO 9
8690 IF N(K9)=1 THEN CHN=CHN+1:CHS(CHN)=K9
8700 NEXT K9
8710 ' kakutei_syori
8720 TY=TTY: FH1=0:GOSUB *TECH2
8730 ' M_modosi
8740 FOR K9=1 TO GK5
8750 Y=HGN(K9)
8760 FOR K8=1 TO GRK(Y)
8770 IF GRTY(Y)=1 THEN TA=GRI(Y):YO=GRIT(Y,K8) ELSE TA=GRIT(Y,K8): YO=GRI(Y)
8780 M(TA,YO)=0
8790 NEXT K8
8800 NEXT K9
8810 IF TUIKA=1 THEN RETURN
8820 NEXT W
8830 RETURN
8840 :::
8900 *GIHOU6
8910 GIHOU=6:BUNRN=14
8920 FOR W=1 TO GKO
8930 TTY=GRTY(W):GI=GRI(W):GJ=GRJ(W)
8940 GRKP=GRP(W)
8950 ' kari_oki
8960 FOR K9=1 TO GRK(W)
8970 IF TTY=1 THEN TA=GI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=GI
8980 M(TA,YO)=GRSU(W,K9)

```

```
8990 NEXT K9
9000 KTY=TTY
9010 ' - - - - type_1 - - - -
9020 FOR K9=1 TO 9: N(K9)=0: NEXT K9
9030 FOR K9=1 TO GRK(W): N(GRSU(W,K9))=1: NEXT K9
9040 ' siyou_su
9050 CHN=0
9060 FOR K9=1 TO 9
9070   IF N(K9)=0 THEN CHN=CHN+1:CHS(CHN)=K9
9080 NEXT K9
9090 '
9100 TY=3 - TTY:II=FNBL(GI)
9110 FH1=0:GOSUB *TECH2
9120 IF TUIKA=1 THEN GOTO *EGSY52
9130 ' - - - - type_2 - - - -
9140 ' siyou_su
9150 CHN=0
9160 FOR K9=1 TO 9
9170   IF N(K9)=1 THEN CHN=CHN+1:CHS(CHN)=K9
9180 NEXT K9
9190 '
9200 TY=3 - TTY
9210 FOR K8=1 TO 3
9220   IF K8<>GJ THEN II=K8: FH1=0:GOSUB *TECH2
9230 NEXT K8
9240 *EGSY52
9250 ' M_modosi
9260 FOR K9=1 TO GRK(W)
9270   IF KTY=1 THEN TA=GI:YO=GRIT(W,K9) ELSE TA=GRIT(W,K9):YO=GI
9280   M(TA,YO)=0
9290 NEXT K9
9300 IF TUIKA=1 THEN RETURN
9310 NEXT W
9320 RETURN
9330 :::
9400 *KOHOSSEN
9410 ' koho_sentak
```

```

9420 IF KOYO<9 THEN STA=KOTA:SYO=KOYO+1 ELSE STA=KOTA+1:SYO=1
9430 FOR TA=STA TO 9
9440 IF TA>STA THEN SYO=1
9450 FOR YO=SYO TO 9
9460 IF M(TA,YO)=0 THEN
9470 ' koho_sagasi
9480 GOSUB *SIYOUSUU
9490 IF PN=7 THEN KOTA=TA:KOYO=YO: GOTO *EKOHOSAGASI
9500 END IF
9510 NEXT YO
9520 NEXT TA
9530 *EKOHOSAGASI
9540 ' kari_oki
9550 KOSU(1)=TSU(1):KOSU(2)=TSU(2)
9560 TA=KOTA:YO=KOYO:SU=KOSU(1): GOSUB *KAKUTEI
9570 RETURN
9580 ::::
9600 *POINTMEAN
9610 ' point_su
9620 FOR I9=0 TO 15: READ PT(I9): NEXT I9
9630 ' keisan
9640 PWA=7.5
9650 FOR I9=0 TO 15: PWA=PWA+PT(I9)*BR(I9): NEXT I9
9660 FOR I9=1 TO 6: PWA=PWA+I9*GRKP(I9): NEXT I9
9670 PHE=PWA/81
9680 ' hyouji
9690 LOCATE 5,22: PRINT "Point_Mean=";PHE
9700 RETURN
9710 :::
9720 ' -- Point_data --
9730 DATA 0,4,2,4,11.5,17,24,32,2,4,11.5,17,24,4,5,4

```

6.5 上にあげた評価値による135題の結果はつぎのようにまとめられる(6.3で述べた基準から、レベル3の6題とレベル4の7題は除いた)。

1) レベル3	2) レベル4	3) レベル5
・問題数: 44	・問題数: 45	・問題数: 46
・平均値: 1.954	・平均値: 2.597	・平均値: 2.795

・標準偏差：0.334      ・標準偏差：0.393      ・標準偏差：0.465

・度数分布表：

以上	未満	度数	以上	未満	度数	以上	未満	度数
	～ 1.5	6	1.75	～ 2	4	2.0	～ 2.4	10
1.5	～ 1.75	6	2	～ 2.25	5	2.4	～ 2.8	18
1.75	～ 2	10	2.25	～ 2.5	6	2.8	～ 3.2	9
2	～ 2.25	15	2.5	～ 2.75	14	3.2	～ 3.6	6
2.25	～	7	2.75	～ 3	9	3.6	～	3
			3	～	7			

これから、

[基準Ⅱ] 1) レベル4以上であるためには、つぎのいずれかの条件が成り立つこと：

- ① 5-空きマス技法以上か技法4～7が必要、
- ② 4-空きマス技法を必要とし、かつ評価値が2.25以上である。

2) レベル5であるためには、5-空きマス技法以上か技法4～7が必要。

・基準2)の根拠：

レベル5であるためには、その評価値がレベル4の平均値を超えた切りの良い値2.6以上となることが最小限みたして欲しい条件であろう（約28%がこの条件をみたさない）。ところでレベル5には4-空きマス技法で解ける問題が12題含まれているが、そのうち9題がこの条件をみたさないからである。

(注意6.3) この節で用いた方法では、レベル4とレベル5を分ける適当な仕切りの値は見つげられない。例えば、5-空きマス技法で解ける問題に関して、

- ・レベル4(10題)：2.623, 2.593, 2.864, 2.549, 2.080, 2.593, 2.278, 2.457, 1.932, 3.019 (平均値=2.499 標準偏差=0.331)
- ・レベル5(8題)：2.481, 2.444, 2.951, 2.796, 2.809, 3.241, 2.198, 2.383 (平均値=2.663 標準偏差=0.345)

であるが、等平均値の検定を危険率5%で行うと、 $t$ -値=1.026, 5%値=2.12より、 $1.026 < 2.12$ だから、等平均値である仮説が棄却できない。

## 7. 今後の課題

6節の議論は、問題が本来持っている難しさややっかいを基にして基準を作ろうとしたものであった。しかし、注意6.3の結果からは、この本来持っている難しさややっかいだけでは難



易度の基準化がうまくいかないことを示している。

従って、人間が解くときの手間とか面倒くささを反映したものを、さらに追加させなければならない。例えば、4-空きマス技法が使われるにしても、適用対象が1組しかないものと、5組もあってその中の1組だけから表示数が見つけれられる場合とでは、難易に対する評価が当然違ってくるはずである。

## 文 献

- [1] 佐藤金吾, イラストパズルを解くコンピュータプログラム, 法政大学多摩研究報告, 15 (2000).
- [2] 『ナンバープレイス②』(1996年 パズルポシェット 日本文芸社)
- [3] 『ナンバープレイス③』(1996年 パズルポシェット 日本文芸社)
- [4] 『ナンプレメイト』(2000年 10月号 マガジン・マガジン社)
- [5] 『ナンプレメイト』(2000年 12月号 マガジン・マガジン社)