

Morio Kikuchi

Abstract :

Developing a regular polyhedron on a plane, setting discrete coordinates on the development and applying a boundary condition of regular polyhedron to it, we realize a symmetrical graphics.

1. Hexahedron

Figure 1 is a development of hexahedron. The number is the number of the vertex and the following are combinations of numbers of vertexes, two sides in correspondence.

- 1, 6 ; 2, 5 ; 3, 7 ; 13, 11 ; 14, 10 ; 8, 12
- 1-2, 6-5 ; 1-3, 6-7 ; 2-4, 5-4 ; 3-8, 7-12 ; 13-14, 11-10 ; 13-8, 11-12 ; 14-9, 10-9

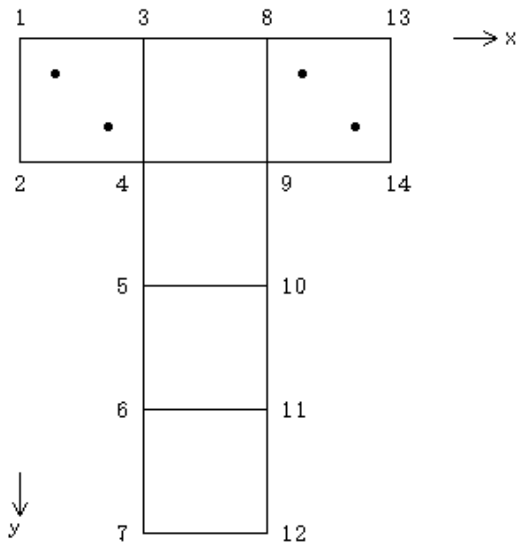


Figure 1

In Figure 1, if we place seed points on square 1-3-4-2 and square 8-13-14-9, a symmetrical graphics is got even if we adopt four rectangles like Figure 2 instead of four squares. In the figure, n and m are as follows: $n = 8$, $m = 12$.

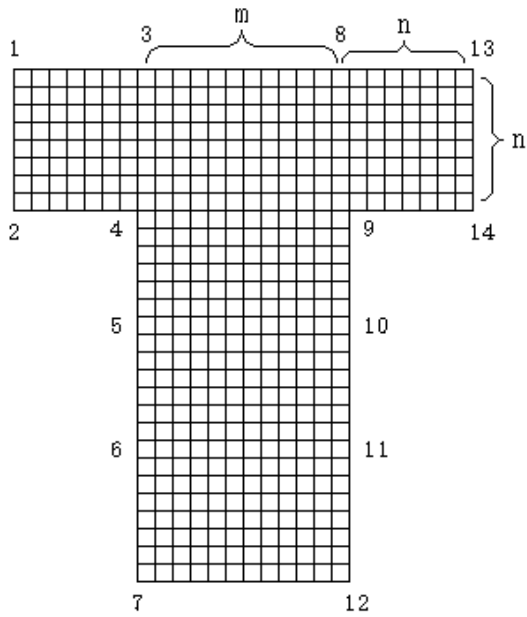


Figure 2

When we want to get odd tunnel in the figure of Figure 2, there is a restriction against n and m . If the number of seed points is made 2+2 like Figure 7-(a), (b), n and m become as follows:

- n : even, m : even, phase : cross or parallel
- n : even, m : odd, phase : cross

cross and parallel represent the placements like Figure 7-(a), (b) respectively. If Figure 7-(a) is developed, Figure 1 is got.

If the number of seed points is made 4+4 like Figure 7-(c), n and m become as follows:

- n : even, m : even

2. Neighborhood view

We show examples of neighborhood view in Figure 1 in the case that there is a jump pixel. In hexahedron, we use square pixel and assume the maximum number of neighborhoods to be 4. We assume n to be the number of pixels of a side of hexahedron.

In a side, we place 4 pixels in 4 directions.

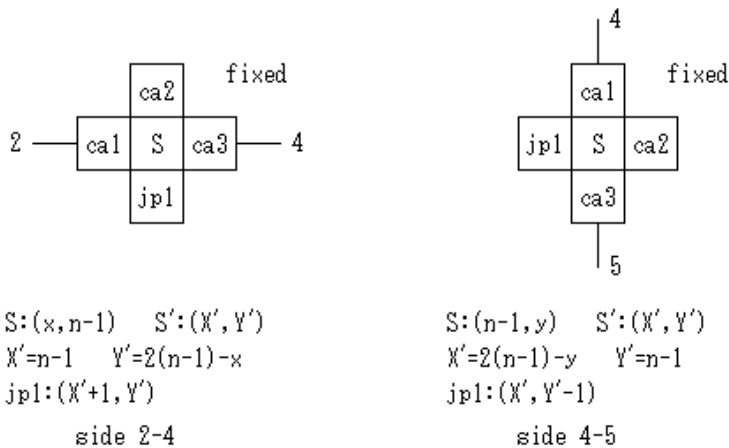


Figure 3

In a vertex, we place 3 pixels in 3 directions.

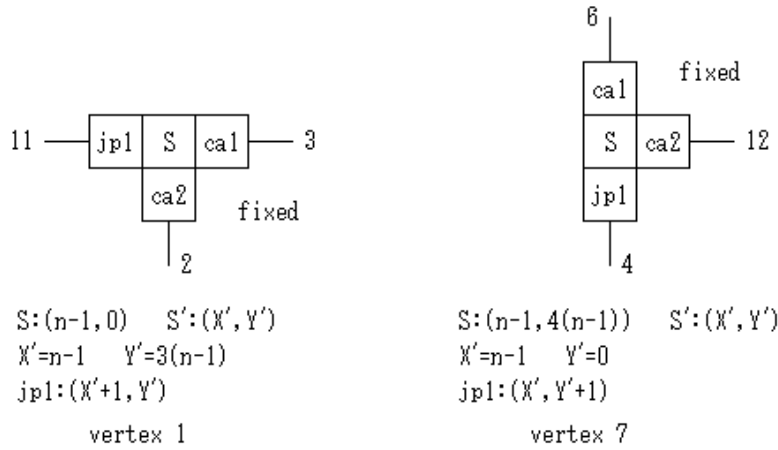


Figure 4

3. Painting algorithm

We show the two painting algorithms in the case that pixel is square and the number of neighborhoods is 4.

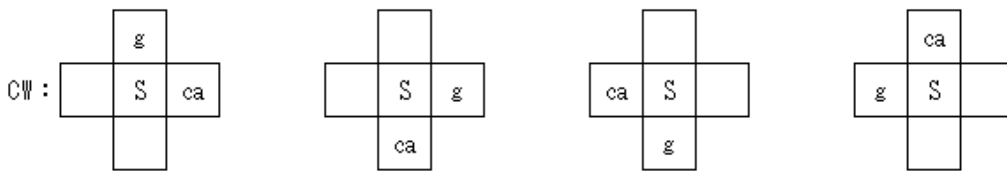


Figure 5

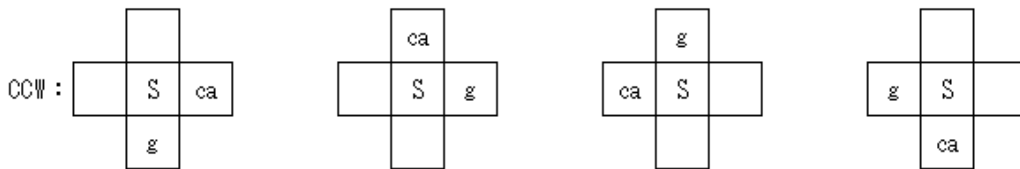


Figure 6

4. Symmetrical axis

In Figure 7, the symmetrical axis goes through a face. Seed points are placed on two minimum convergence surfaces. In this case, because secondary convergence surfaces exist, other placements of seed points is possible.

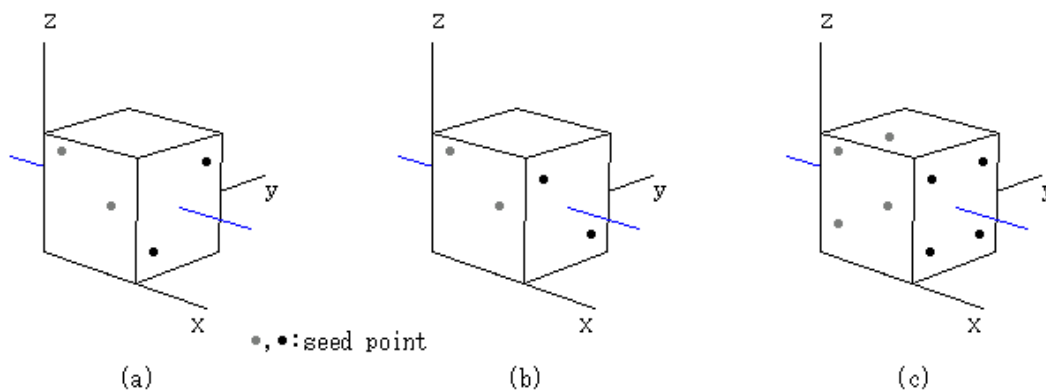


Figure 7

In Figure 8, the symmetrical axis goes through a vertex. Because three painting points fight for one vertex through which the symmetrical axis goes, we close the corresponding vertexes beforehand. They are 3, 7, 10, 14 in Figure 1 and P, Q in Figure 8.

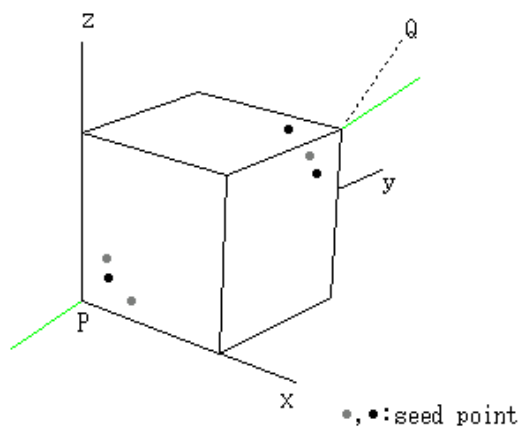


Figure 8

5. Processing of hexahedron

If the inside of a hexahedron is cut, Figure 9 is got. By the cut, the each surface becomes \square type circuit. It is called outer face and the number of pixels of the road width is assumed to be n_w . The face which appears inside the hexahedron is called inner face.

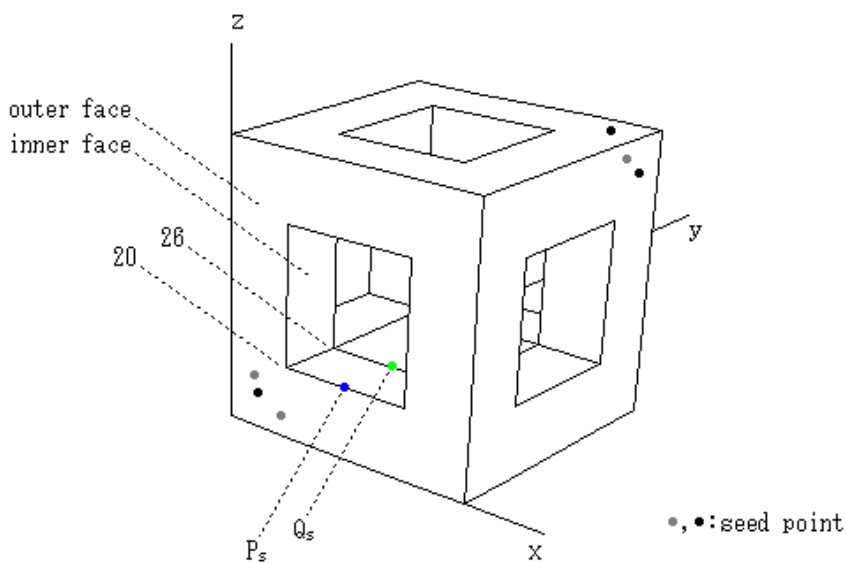


Figure 9

If Figure 9 is developed, Figure 10 is got. The left figure is a group of outer faces and the right figures are a group of inner faces.

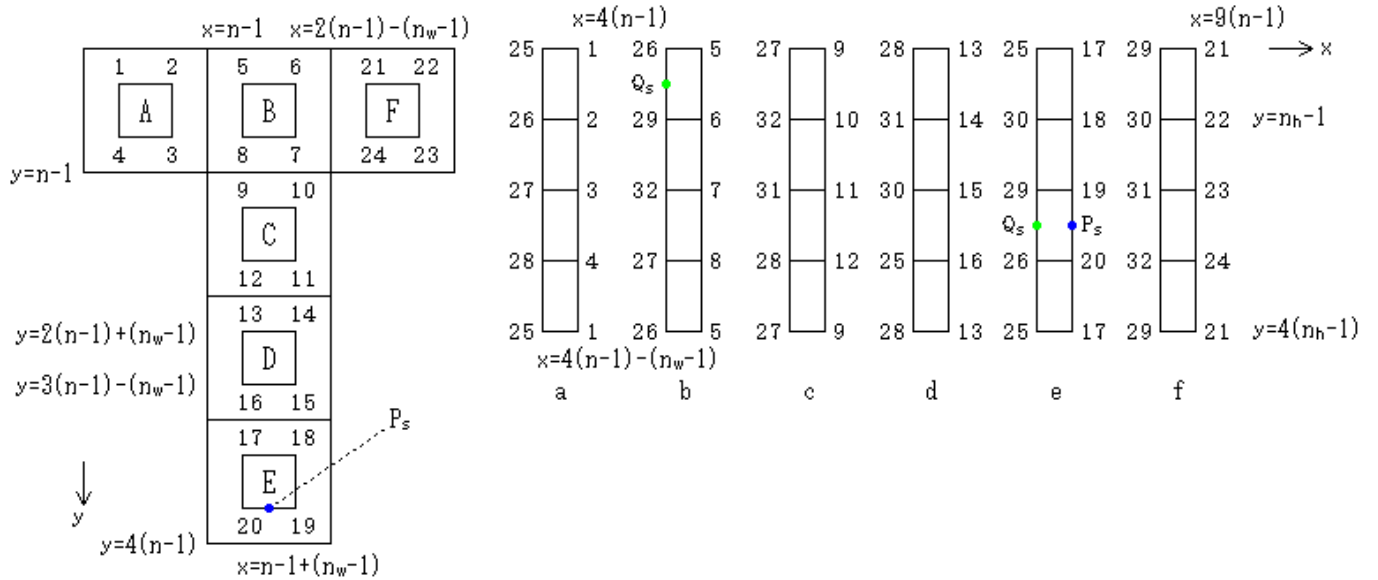


Figure 10

The number of vertex of outer face is 1 ~ 24 and the number of vertex of inner face is 25 ~ 32. For example, one inner face a (small letter) connects with outer face A (capital letter) and other nine inner faces. Besides, the upper line and lower line of one inner face column connect. In the figure, n_h is the number of pixels in the y direction of one inner face and assuming n to be the number of pixels of a side of hexahedron, it is as follows:

- $n_h = n - 2(n_w - 1)$

In vertex and side, because the number of neighborhoods exceeds 4, we use logical angle method in the choice of pixel to paint. We show the elements of logical angle method at P_s in Figure 9 using angles(deg.) of direction to which a pixel exists.

- (1) 180, 270, 360 (on P_s in E) ; 180 (on P_s in e)

The elements of logical angle method at Q_s in Figure 9 are

- (2) -90, 0, 90 (on Q_s in e) ; 0 (on Q_s in b)

The elements of logical angle method at vertex 20 in Figure 9 are

- (3) 90, 180, 270, 360 (on vertex 20 in E) ; 180 (on vertex 20 in e)

The elements of logical angle method at vertex e-20 in Figure 10 are

- (4) 90, 180, 270 (on vertex 20 in e) ; 180, 270 (on vertex 20 in E)

The elements of logical angle method at vertex 26 in Figure 9 are

- (5) -90, 0, 90 (on vertex 26 in e) ; 0, 90 (on vertex 26 in b) ; 0 (on vertex 26 in a)

In order for the graphs to be made symmetrically, we must do the choice of pixel to paint in an unified

manner. For example, in (4), (5), as shown at the right of "20:", "26:" in Figure 11 respectively, when the right screw is rotated, in the image like the head of the right screw rotates, we choose pixels. As a result of the choice, pixels are chosen counterclockwise on the each face.

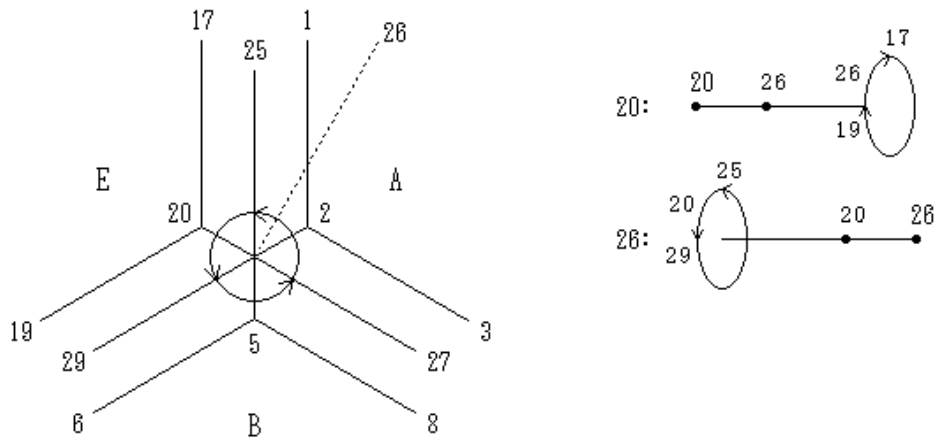


Figure 11

Seed points in Figure 9 are the same as Figure 8, however, because the inside is cut, we close not only the two vertexes in Figure 8 but also the vertexes 26, 31 in Figure 10.

6. Processing of tetrahedron

If the inside of a tetrahedron is cut, Figure 12 is got and if Figure 12 is developed, Figure 13 is got. The technique to made a symmetrical graphics is quite the same as Figure 9, 10 except the combination of hexagonal pixel and oblique coordinate system and the relevant matters.

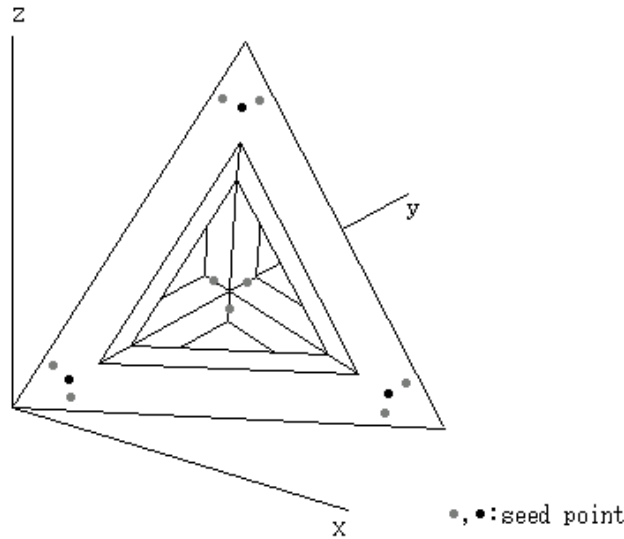


Figure 12

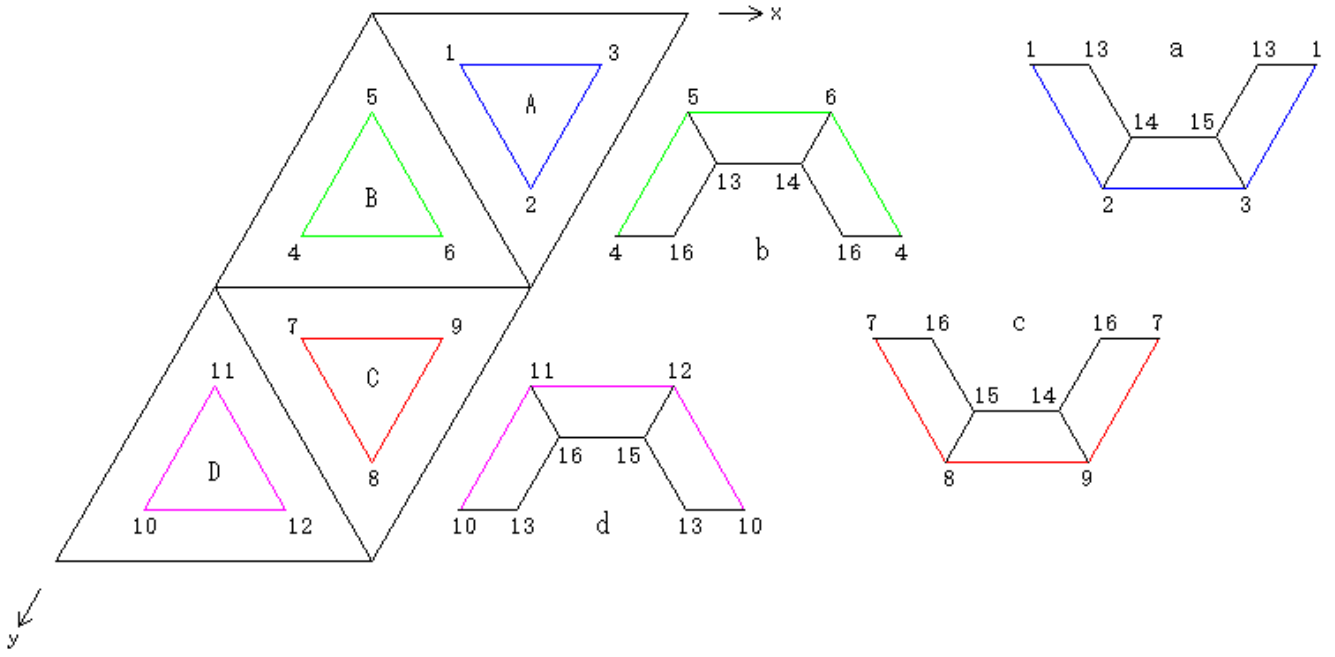


Figure 13

If we assume the number of pixels of the road width of the circuit to be n_w , assuming n to be the number of pixels of a side of tetrahedron, the number of pixels of a side of one inner face is as follows:

- $n_{ho} = n - 3(n_w - 1)$
- $n_{hi} = n_{ho} - (n_w - 1)$

n_{ho} is the number of pixels of a side which borders on an outer face and n_{hi} is the number of pixels of a side which borders on other inner faces.

The following are the coordinates of each vertex.

- A-1 : $(2(n_w - 1), n_w - 1)$, A-2 : $(n - 1 - (n_w - 1), n - 1 - 2(n_w - 1))$
- B-5 : $(n_w - 1, 2(n_w - 1))$, B-6 : $(n - 1 - 2(n_w - 1), n - 1 - (n_w - 1))$
- C-7 : $(2(n_w - 1), n - 1 + n_w - 1)$, C-8 : $(n - 1 - (n_w - 1), 2(n - 1) - 2(n_w - 1))$
- D-11 : $(n_w - 1, n - 1 + 2(n_w - 1))$, D-12 : $(n - 1 - 2(n_w - 1), 2(n - 1) - (n_w - 1))$

- a-1-left : $(2(n - 1) + n_w - 1, n_w - 1)$
- a-3 : $(2(n - 1) + n_w - 1 + 2(n_{ho} - 1), n_w - 1 + n_{ho} - 1)$
- a-13-left : $(2(n - 1) + 2(n_w - 1), n_w - 1)$
- a-14 : $(2(n - 1) + n_w - 1 + n_{ho} - 1, n_w - 1 + n_{hi} - 1)$
- a-15 : $(2(n - 1) + n_w - 1 + n_{ho} - 1 + n_{hi} - 1, n_w - 1 + n_{hi} - 1)$

- b-4-left : $(n - 1 + n_w - 1, n - 1 - (n_w - 1))$
- b-6 : $(n - 1 + n_w - 1 + n_{ho} - 1, 2(n_w - 1))$
- b-4-right : $(n - 1 + n_w - 1 + 2(n_{ho} - 1), n - 1 - (n_w - 1))$
- b-16-right : $(n - 1 + n_w - 1 + n_{ho} - 1 + n_{hi} - 1, n - 1 - (n_w - 1))$
- b-13 : $(n - 1 + 2(n_w - 1), 3(n_w - 1))$

- c-7-left : $(2(n - 1) + n_w - 1, n - 1 + n_w - 1)$

- c-9 : $(2(n - 1) + n_w - 1 + 2(n_{ho} - 1), n - 1 + n_w - 1 + n_{ho} - 1)$
- c-16-left : $(2(n - 1) + 2(n_w - 1), n - 1 + n_w - 1)$
- c-15 : $(2(n - 1) + n_w - 1 + n_{ho} - 1, n - 1 + n_w - 1 + n_{hi} - 1)$
- c-14 : $(2(n - 1) + n_w - 1 + n_{ho} - 1 + n_{hi} - 1, n - 1 + n_w - 1 + n_{hi} - 1)$

- d-10-left : $(n - 1 + n_w - 1, 2(n - 1) - (n_w - 1))$
- d-12 : $(n - 1 + n_w - 1 + n_{ho} - 1, n - 1 + 2(n_w - 1))$
- d-10-right : $(n - 1 + n_w - 1 + 2(n_{ho} - 1), 2(n - 1) - (n_w - 1))$
- d-13-right : $(n - 1 + n_w - 1 + n_{ho} - 1 + n_{hi} - 1, 2(n - 1) - (n_w - 1))$
- d-16 : $(n - 1 + 2(n_w - 1), n - 1 + 3(n_w - 1))$

7. Assignment

- Complete neighborhood views on vertexes and sides of the figure of Figure 1.
- Get neighborhood views on vertexes and sides of the figure of Figure 10.
- Get neighborhood views on vertexes and sides of the figure of Figure 13.

8. Concrete example

Figure 14 is a symmetrical graphics by Figure 9 and the following are data of program.

- SP(1 × 6):1(sA)+1(sE)+1(sB)+1(sC)+1(sF)+1(sD)
- $n = 16$

- coordinates of painting number 1 : painting point a: $(n - 3, 2)$ 、 painting point b: $(n + 1, 4(n - 1) - 2)$ 、 painting point c: $(n + 1, 2)$
- coordinates of painting number 1 : painting point d: $(2(n - 1) - 2, 2(n - 1) - 2)$ 、 painting point e: $(3(n - 1) - 2, n - 3)$ 、 painting point f: $(2(n - 1) - 2, 2(n - 1) + 2)$
- coordinates of painting number 2 : painting point a: $\Delta y = -1$ 、 painting point b: $\Delta y = 1$ 、 painting point c: $\Delta x = -1$
- coordinates of painting number 2 : painting point d: $\Delta x = 1$ 、 painting point e: $\Delta x = 1$ 、 painting point f: $\Delta y = -1$

If N1 is set to 0, a symmetrical graphics by Figure 8 is got.

```
#define N1 0          /* nw>=2 */
```

If painting number 2 is finished at the first graphics, program pauses. Press Esc key.

- choice of CW, CCW : the same as the first
- painting algorithm : logical angle method
- painting timing : immediate painting
- push to stack : the same as the first

Array which is used for painting is initialized as follows:

- target pixel : 15
- wall pixel : 0

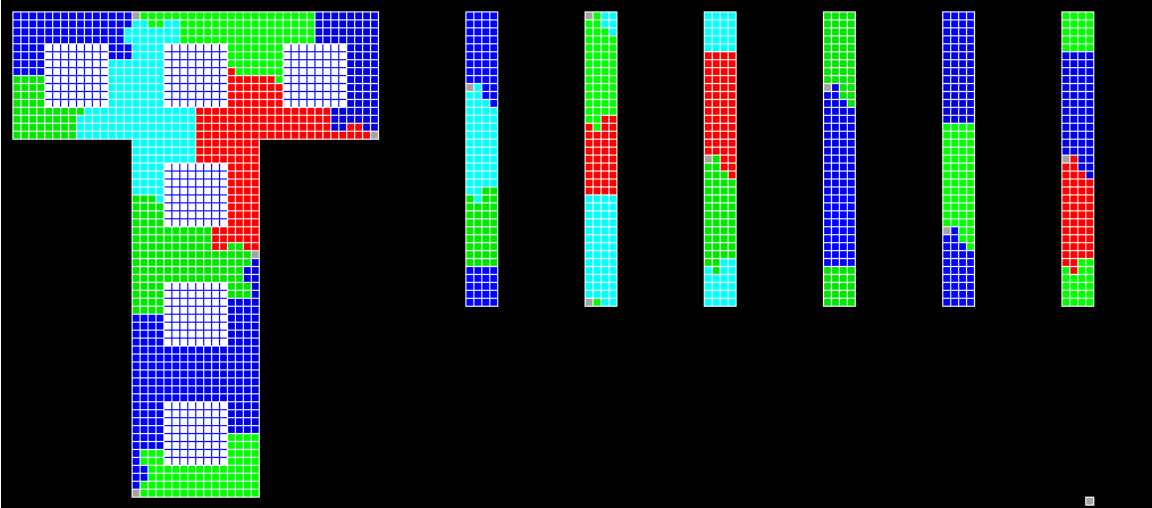


Figure 14

セルラーオートマトングラフィクス (6)

菊池盛雄

アブストラクト :

正多面体を平面上に展開し、この展開図形に離散座標を設定し、正多面体の境界条件を適用して対称なグラフィクスを実現します。

1. 正6面体

図1は正六面体の展開図です。数字は頂点の番号であり、以下は対応する頂点、辺の組合せです。

- 1, 6 ; 2, 5 ; 3, 7 ; 13, 11 ; 14, 10 ; 8, 12
- 1-2, 6-5 ; 1-3, 6-7 ; 2-4, 5-4 ; 3-8, 7-12 ; 13-14, 11-10 ; 13-8, 11-12 ; 14-9, 10-9

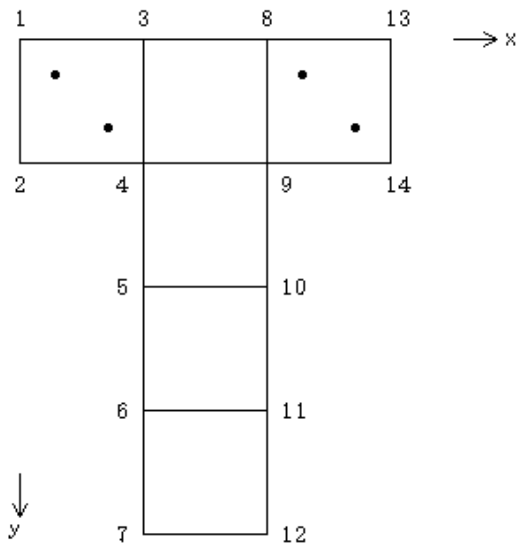


図1

図1において、正方形1-3-4-2と正方形8-13-14-9にシードポイントを配置する場合は、間にある正方形をその辺を延長して図2のような長方形にしても対称なグラフィクスが得られます。図においては $n = 8$ 、 $m = 12$ となっています。

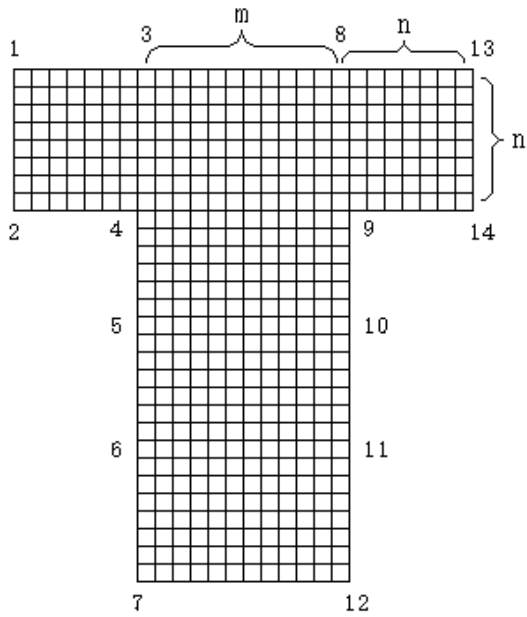


図 2

もし、図 2 の図形において奇トンネルを得たい場合には n 、 m には制限があります。シードポイントの数を図 7-(a), (b) のように $2+2$ にすれば n 、 m は以下ようになります。

- n : even, m : even, phase : cross or parallel
- n : even, m : odd, phase : cross

cross、parallel は各々図 7-(a), (b) のような配置を指します。図 7-(a) を展開すれば図 1 が得られます。シードポイントの数を図 7-(c) のように $4+4$ にすれば n 、 m は以下ようになります。

- n : even, m : even

2. 近傍図

ジャンプピクセルが存在する場合の図 1 における近傍図の例を示します。正六面体においては正方形ピクセルを用い、最大近傍数は 4 とします。 n を正六面体の辺のピクセル数とします。辺においては四方向のピクセルを配置します。

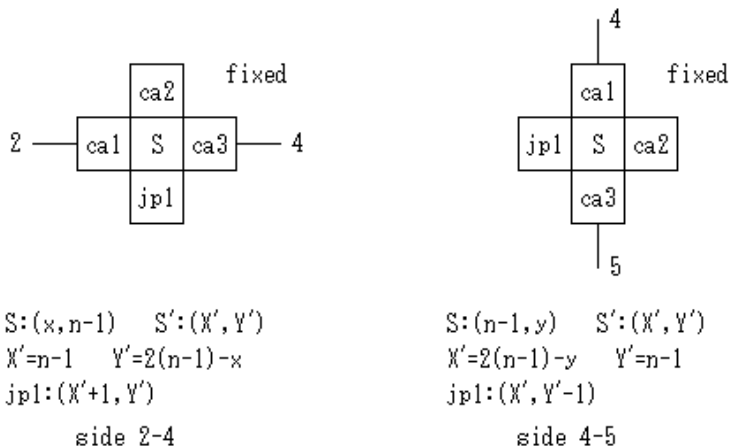


図 3

頂点においては三方向のピクセルを配置します。

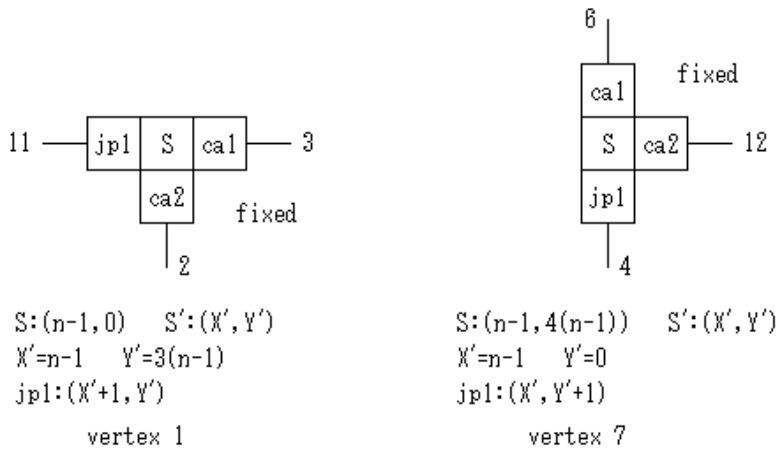


図 4

3. 塗りつぶしアルゴリズム

ピクセルが正方形で近傍数が 4 の場合の塗りつぶしアルゴリズムを示します。

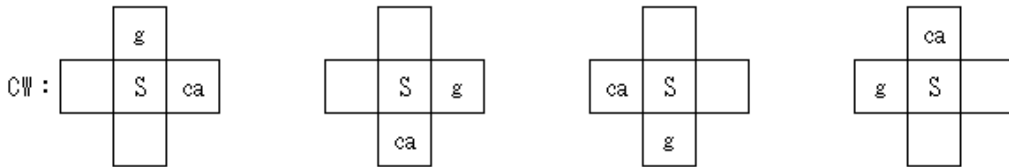


図 5

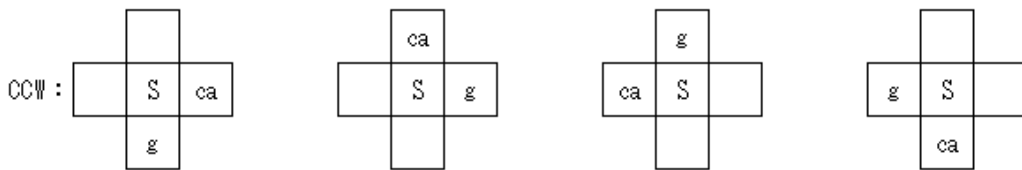


図 6

4. 対称軸

図 7 は対称軸が面を通る場合です。シードポイントは最小収束面に配置してあります。この場合は二次収束面が存在するので他のシードポイントの配置も可能です。

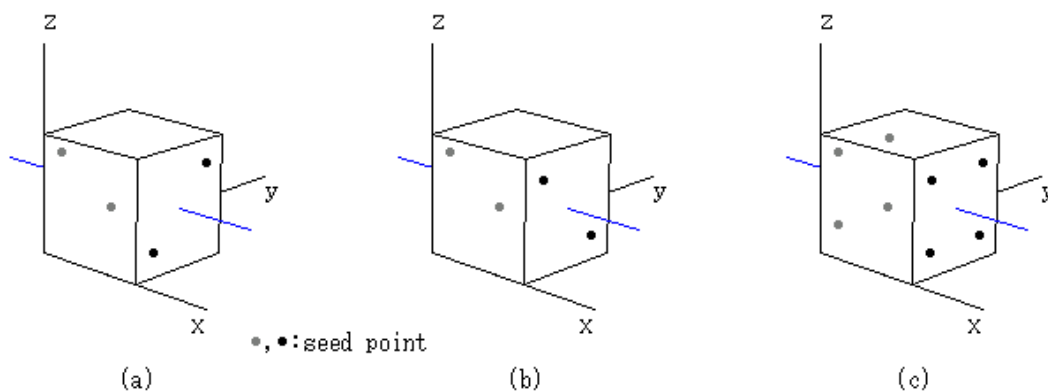


図 7

図 8 は対称軸が頂点を通る場合です。図のようなシードポイントの配置を行うと三つの塗点が対称軸が通る頂点を奪い合うので、あらかじめ対応する頂点を閉じておきます。図 1 では 3、7、10、14 であり、図 8 では P、Q です。

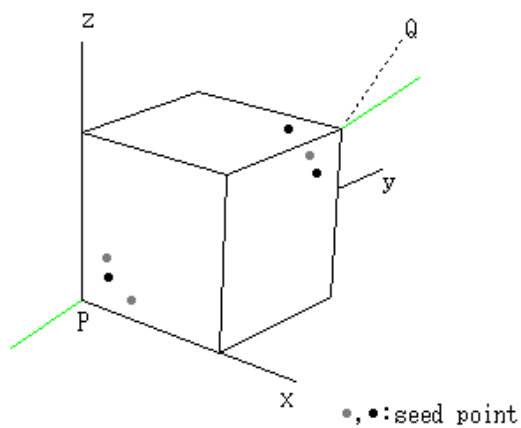


図 8

5. 正六面体の加工

正六面体の内部ををくり抜けば図 9 が得られます。くり抜きによって正六面体の各面は回型サーキットになります。これを outer face と称し、この道幅のピクセル数を n_w とします。正六面体の内部に生じた面を inner face と称します。

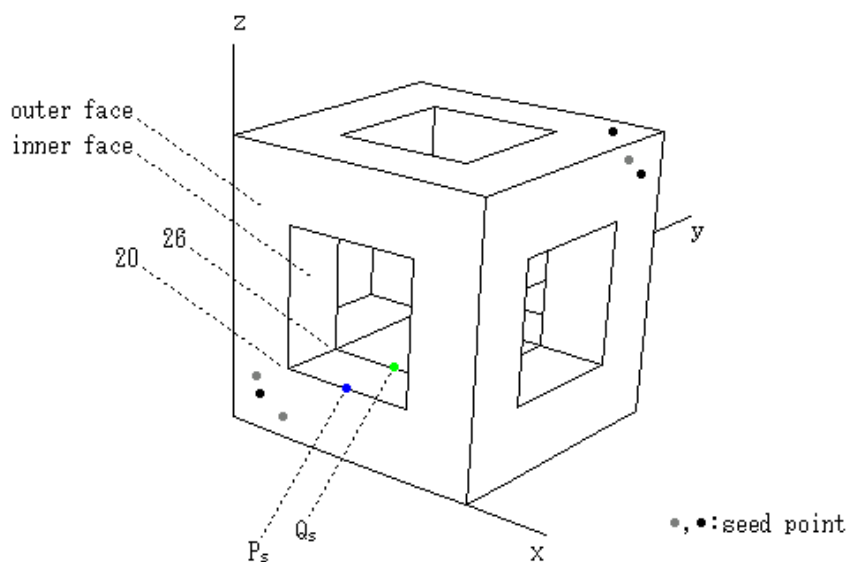


図 9

図 9 を展開すると図 10 が得られます。左図が outer face の集まり、右図が inner face の集まりです。

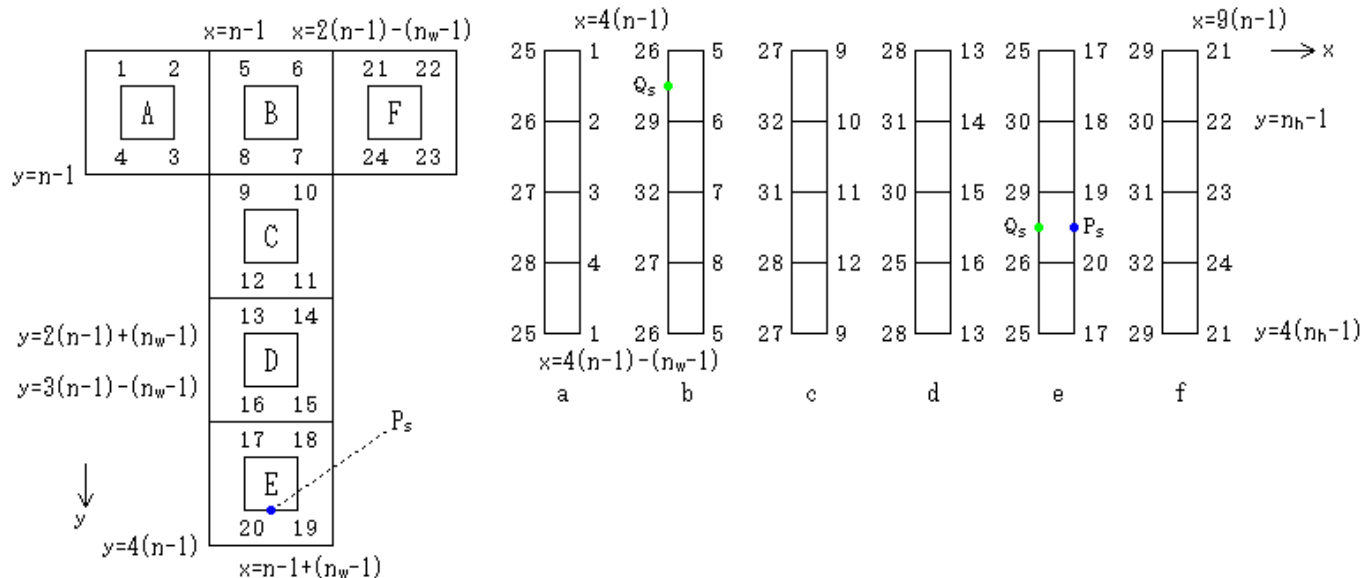


図 10

outer face の頂点の番号は 1 ~ 24 であり、inner face の頂点の番号は 25 ~ 32 です。たとえば、ひとつの inner face a (小文字) は outer face A (大文字) および他の九つの inner face とつながっています。また、ひとつの inner face 列は上と下がつながっています。図において n_h はひとつの inner face の y 方向のピクセル数であり、 n を正六面体の辺のピクセル数として以下のようにになっています。

- $n_h = n - 2(n_w - 1)$

頂点や辺では近傍数が 4 を越えるので塗りつぶすピクセルの選択には論理角度法を用います。図 9 の P_s における論理角度法の要素をピクセルの存在する方向の角度 (deg.) を用いて示します。

- (1) 180, 270, 360 (on P_s in E) ; 180 (on P_s in e)

図 9 の Q_s における論理角度法の要素は

- (2) -90, 0, 90 (on Q_s in e) ; 0 (on Q_s in b)

図 9 の頂点 20 における論理角度法の要素は

- (3) 90, 180, 270, 360 (on vertex 20 in E) ; 180 (on vertex 20 in e)

図 10 の頂点 e-20 における論理角度法の要素は

- (4) 90, 180, 270 (on vertex 20 in e) ; 180, 270 (on vertex 20 in E)

図 9 の頂点 26 における論理角度法の要素は

- (5) -90, 0, 90 (on vertex 26 in e) ; 0, 90 (on vertex 26 in b) ; 0 (on vertex 26 in a)

グラフィクスが対称に描かれるためには塗りつぶすピクセルの選択を統一的行わなければなりません。たとえば (4)、(5) では各々図 11 の "20:"、"26:" の右に示されるように、右ねじを回したときにねじの頭が回転するようなイメージでピクセルを選択します。その結果、各面内では反時計回りにピクセル

が選択されます。

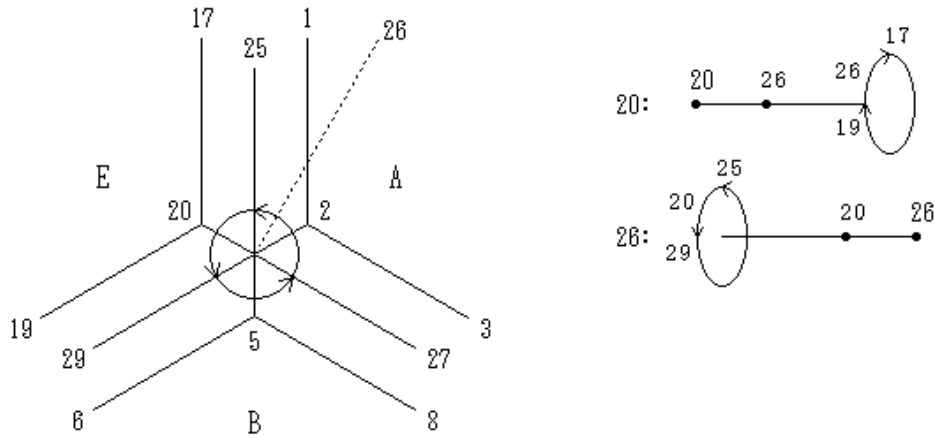


図 11

図 9 におけるシードポイントは図 8 と同じですが、内部がくり抜かれているので図 8 における二つの頂点だけでなく図 10 における頂点 26、31 も閉鎖します。

6. 正四面体の加工

正四面体の内部ををくり抜けば図 12 が得られ、図 12 を展開すると図 13 が得られます。対称なグラフィクスを描く手法は、正六角形ピクセルと斜交座標の組み合わせおよびこの関連を除いて、図 9、10 と全く同じです。

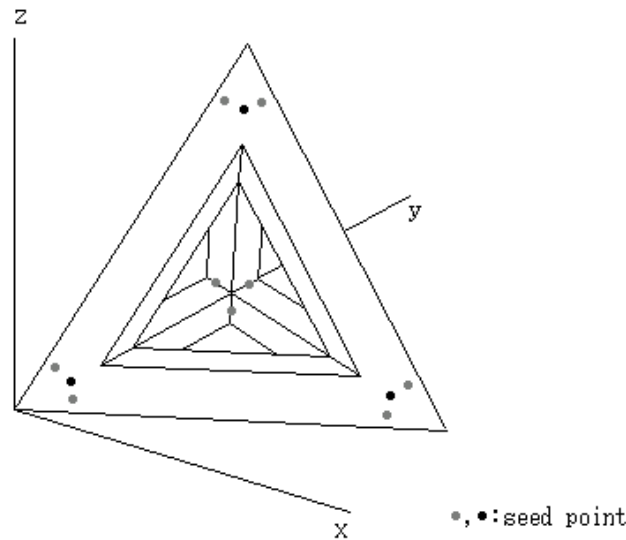


図 12

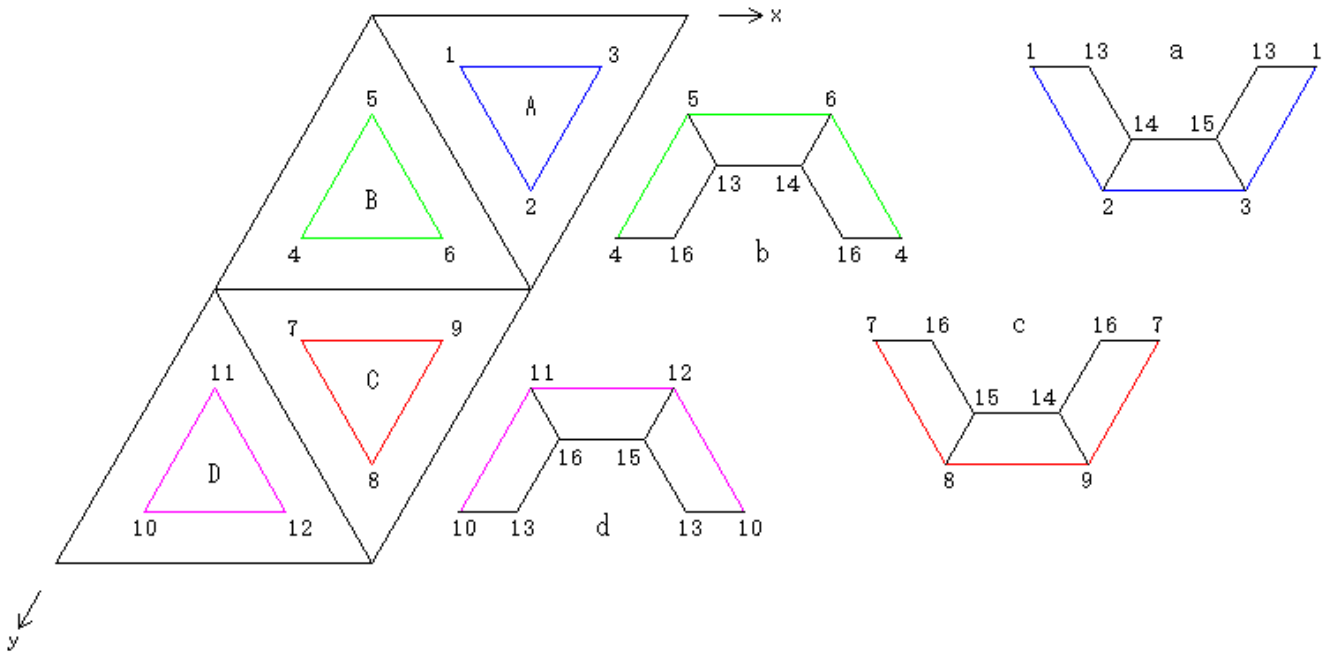


図 13

サーキットの道幅のピクセル数を n_w とすれば、ひとつの inner face の辺のピクセル数は、 n を正四面体の辺のピクセル数として以下のようにになっています。

- $n_{ho} = n - 3(n_w - 1)$
- $n_{hi} = n_{ho} - (n_w - 1)$

n_{ho} は outer face と接している辺のピクセル数であり、 n_{hi} は他の inner face と接している辺のピクセル数です。

各頂点の座標を以下に示します。

- A-1 : $(2(n_w - 1), n_w - 1)$, A-2 : $(n - 1 - (n_w - 1), n - 1 - 2(n_w - 1))$
- B-5 : $(n_w - 1, 2(n_w - 1))$, B-6 : $(n - 1 - 2(n_w - 1), n - 1 - (n_w - 1))$
- C-7 : $(2(n_w - 1), n - 1 + n_w - 1)$, C-8 : $(n - 1 - (n_w - 1), 2(n - 1) - 2(n_w - 1))$
- D-11 : $(n_w - 1, n - 1 + 2(n_w - 1))$, D-12 : $(n - 1 - 2(n_w - 1), 2(n - 1) - (n_w - 1))$

- a-1-left : $(2(n - 1) + n_w - 1, n_w - 1)$
- a-3 : $(2(n - 1) + n_w - 1 + 2(n_{ho} - 1), n_w - 1 + n_{ho} - 1)$
- a-13-left : $(2(n - 1) + 2(n_w - 1), n_w - 1)$
- a-14 : $(2(n - 1) + n_w - 1 + n_{ho} - 1, n_w - 1 + n_{hi} - 1)$
- a-15 : $(2(n - 1) + n_w - 1 + n_{ho} - 1 + n_{hi} - 1, n_w - 1 + n_{hi} - 1)$

- b-4-left : $(n - 1 + n_w - 1, n - 1 - (n_w - 1))$
- b-6 : $(n - 1 + n_w - 1 + n_{ho} - 1, 2(n_w - 1))$
- b-4-right : $(n - 1 + n_w - 1 + 2(n_{ho} - 1), n - 1 - (n_w - 1))$
- b-16-right : $(n - 1 + n_w - 1 + n_{ho} - 1 + n_{hi} - 1, n - 1 - (n_w - 1))$
- b-13 : $(n - 1 + 2(n_w - 1), 3(n_w - 1))$

- c-7-left : $(2(n - 1) + n_w - 1, n - 1 + n_w - 1)$

- c-9 : $(2(n-1) + n_w - 1 + 2(n_{ho} - 1), n - 1 + n_w - 1 + n_{ho} - 1)$
- c-16-left : $(2(n-1) + 2(n_w - 1), n - 1 + n_w - 1)$
- c-15 : $(2(n-1) + n_w - 1 + n_{ho} - 1, n - 1 + n_w - 1 + n_{hi} - 1)$
- c-14 : $(2(n-1) + n_w - 1 + n_{ho} - 1 + n_{hi} - 1, n - 1 + n_w - 1 + n_{hi} - 1)$
- d-10-left : $(n - 1 + n_w - 1, 2(n-1) - (n_w - 1))$
- d-12 : $(n - 1 + n_w - 1 + n_{ho} - 1, n - 1 + 2(n_w - 1))$
- d-10-right : $(n - 1 + n_w - 1 + 2(n_{ho} - 1), 2(n-1) - (n_w - 1))$
- d-13-right : $(n - 1 + n_w - 1 + n_{ho} - 1 + n_{hi} - 1, 2(n-1) - (n_w - 1))$
- d-16 : $(n - 1 + 2(n_w - 1), n - 1 + 3(n_w - 1))$

7. 課題

- 図 1 の図形の頂点、辺に関する近傍図を完成させてください。
- 図 10 の図形の頂点、辺に関する近傍図を求めてください。
- 図 13 の図形の頂点、辺に関する近傍図を求めてください。

8. 具体例

図 14 は図 9 による対称グラフィクスであり、以下はプログラムのデータです。

- SP(1 × 6):1(sA)+1(sE)+1(sB)+1(sC)+1(sF)+1(sD)
- $n = 16$
- 塗番号 1 の座標 : 塗点 a: $(n - 3, 2)$ 、塗点 b: $(n + 1, 4(n - 1) - 2)$ 、塗点 c: $(n + 1, 2)$
- 塗番号 1 の座標 : 塗点 d: $(2(n - 1) - 2, 2(n - 1) - 2)$ 、塗点 e: $(3(n - 1) - 2, n - 3)$ 、塗点 f: $(2(n - 1) - 2, 2(n - 1) + 2)$
- 塗番号 2 の座標 : 塗点 a: $\Delta y = -1$ 、塗点 b: $\Delta y = 1$ 、塗点 c: $\Delta x = -1$
- 塗番号 2 の座標 : 塗点 d: $\Delta x = 1$ 、塗点 e: $\Delta x = 1$ 、塗点 f: $\Delta y = -1$

N1 を 0 にすると図 8 による対称グラフィクスが得られます。

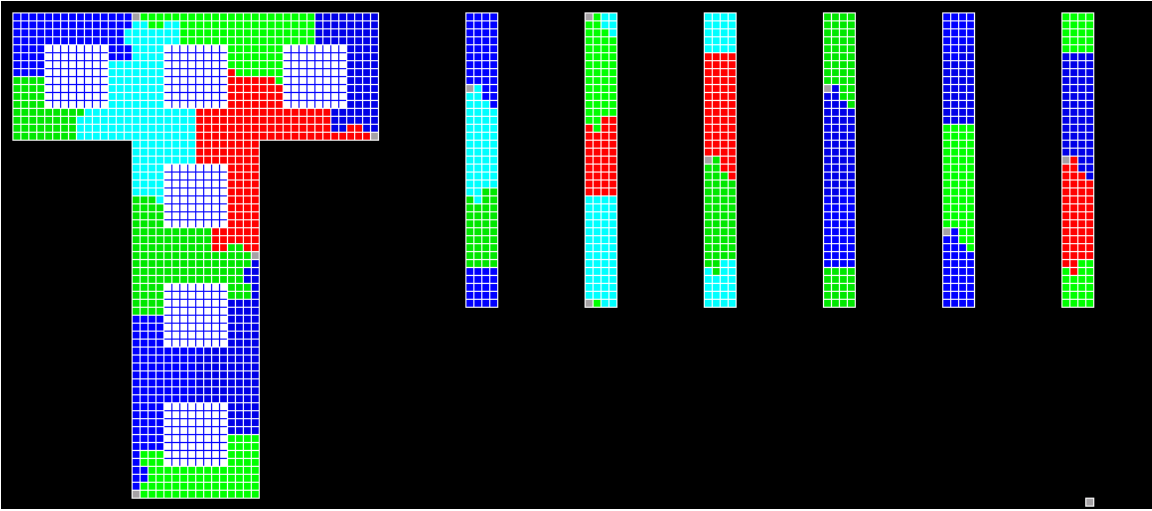
```
#define N1 0          /* nw>=2 */
```

最初のグラフィクスで塗番号 2 が終了するとプログラムが一時停止します。Esc キーを押してください。

- CW、CCW の選択 : 初回と同じ
- 塗りつぶしアルゴリズム : 論理角度法
- 塗り方 : 即時塗りつぶし
- スタックへの座標のプッシュ : 初回と同じ

塗りつぶしに用いられる配列は以下のように初期化します。

- ターゲットピクセル : 15
- 壁ピクセル : 0



☒ 14

List 1:cag_6.c

```
/* t2.28.s */
/* 2019 Morio Kikuchi */

#include <windows.h>
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <math.h>

#define VGACOLORS 50
#define GKS GetKeyState
#define ASIZE_MS ((1024*768)/CPMAX)

#define CPMAX /*4*//*8*//*24*//*3*//*6*/6
#define RESO 16 /* n */
#define DSP (-0) /* 3+DSP<=N1 */
#define N1 /*4*/4 /* nw>=2 */
#define N2 (RESO-2*(N1-1)) /* nh */

#if CPMAX==8 || CPMAX==24 || CPMAX==6
#define DIV 1
#elif CPMAX==3
#define DIV 0
#else
#define DIV /*0*//*1*/1 /* CPMAX:4 => 0,1 */
#endif

#if DIV==0
#define CPHALF CPMAX
#else
#define CPHALF (CPMAX/2)
#endif

#define CROSS 0

#define X0 (10)
#define Y0 (10)
#define dyMAX 603
#define GRPH_0_MAX 2000

#define ICEIL(a,b) (((a)+((b)-1))/(b))
```

```

char refill, pauseflag, fieldflag, GRPH;
char charcode, charflag;
int X, Y, X_, Y_, qx, qy, qx_, qy_, qpos, iv[6][5][2][2];
int enX[6+6], enY[6+6], enX_[6+6], enY_[6+6], enSN[6+6];
int algo, combination, drn, ig, PIXSIZE, jmpflag, sn_, sn;
int xt, yt, ssize, std_x, std_y, last_x, last_y, tmp0, tmp1;
long asize=ASIZE_MS;
long rcount[CPMAX], cnt;

/*unsigned */char **pixel;
long fp_mem[CPMAX];

char function, usflag;
unsigned char yorn;
int XRESO, YRESO, WB, DX_FRAME, DY_FRAME, DY_CAPTION;
FILE *fp;

typedef struct {int xx, yy, xx_, yy_, sn;} ss;
ss s; ss rtn[CPMAX][ASIZE_MS];
typedef struct {
unsigned char red, green, blue;} srgb;
srgb irgb[VGACOLORS];
typedef struct {
unsigned long back_; int back, fore;} bf;
bf bfset[] = {{WHITENESS, 15, 0}, {BLACKNESS, 0, 15}};

HINSTANCE hinstance;
HWND hwnd;
HDC hdcdisplay, hdctmp1;
HBITMAP hbitmap1;
HPEN hpen;
HBRUSH hbrush;

void closegraph_(void), initpalette(void), BitBlt_full(void), setup(void),
cleardevice_(char, int, int, int, int), rectangle_(int, int, int, int),
delay_(long), beep(long), kbhit_(void), restore_3(void), initgraph_return(void),
use_subroop(void), keydowns_f2(void), bitblt(char, int, int, int, int, int, int),
arrayreset(void), fwrite_mem(int), fread_mem(int), putpixel_(int, int, int),
check_rcount(void), field(void);
unsigned char subroop(void);
int initgraph_(void), setup_(void), fourfloor_fiveceil(double), random_(int),
getpixel_(int, int, int, int), cag_r(void);
long ftell_mem(int);
double getangle(int, int);

COLORREF PALETTE(int color);

```

```

LRESULT CALLBACK wndproc_by_kbhit_(HWND,UINT,WPARAM,LPARAM);
int wndproc_filer(HWND,UINT,WPARAM,LPARAM);

int main(int argc,unsigned char **argv)
{
long mytime;

if(argc>1 && strcmp(argv[1],"0")==0){
GRPH=0;
if(argc==2) argc=1;else argc=2;
}
else GRPH=1;
WB=1;
refill=1;
/*unlink("cpage.bin");*/

if(initgraph_()==1) return 1;

cleardevice_(1,0,0,XRESO,YRESO);
BitBlt_full();

xt=(3+6)*(RESO-1);yt=4*(RESO-1);
if(setup_()==1) return 1;
arrayreset();
printf(" xt+1:%d\n",xt+1);putpixel(xt,yt+1,8);

if(argc>1) {time(&mytime);srand((unsigned int)mytime);}
else
srand(1);

combination=1;
drn=4;

while(1){
field();
cag_r();
if(refill==0) break;
check_rcount();
printf(" \n");
if(refill==0) break;

if(GRPH){
beep(50);

delay_(6000);

```

```

if(pauseflag==1) {pauseflag=0;use_subroop();}
}/**if(GRPH)**/
if(refill==0) break;
}/**while(1)**/

closegraph_();

return 0;
}/** main **/

void ls_image(char flag,char *file,int x,int y,int dx,int dy)
{
unsigned long xsize,ysize,size;
unsigned long width,height,imagesize;
unsigned long bits,bytesPerPixel,lineSizeDW,lineSize;
HDC hdce,hdc;
HBITMAP hbitmape;
BITMAPFILEHEADER bfh;
BITMAPINFOHEADER bih;
BYTE *gdata;
FILE *fpo,*fpi;

if(flag<=3){ /* save */
if(flag==0){
}
else if(flag==1){
}
else if(flag==2){
}
else if(flag==3){
}
else return;

if((fpo=fopen(file,"wb"))==NULL) {printf("Can't open a file.\n");return;}

width=dx;
height=dy;

bits=/*16*/24/*32*/;
bytesPerPixel=bits/8;
lineSizeDW=bytesPerPixel*width;
lineSizeDW=ICEIL(lineSizeDW,sizeof(long));
lineSize=lineSizeDW*sizeof(long);
imagesize=lineSize*height;

```

```

bfh.bfType=0x4d42;                /* "BM" */
bfh.bfSize=54+imagesize;
bfh.bfReserved1=0;
bfh.bfOffBits=54;
bfh.bfReserved2=0;

bih.biSize=40;
bih.biWidth=width;
bih.biHeight=height;
bih.biPlanes=1;
bih.biBitCount=bits;
bih.biCompression=0;
bih.biSizeImage=imagesize;
bih.biXPelsPerMeter=0;
bih.biYPelsPerMeter=0;
bih.biClrUsed=0;
bih.biClrImportant=0;

if(flag<=1)
/*hdce=CreateCompatibleDC(hdctmp2)*/;
else if(flag==2)
hdce=CreateCompatibleDC(hdctmp1);
else{
hdc=CreateDC("DISPLAY",NULL,NULL,NULL);
hdce=CreateCompatibleDC(hdc);
}

hbitmap=CreateDIBSection(hdce,(LPBITMAPINFO)&bih,DIB_RGB_COLORS,&gdata,NULL,0);
SelectObject(hdce,hbitmap);

if(flag<=1)
/*BitBlt(hdce,0,0,dx,dy,hdctmp2,x,y,SRCCOPY)*/;
else if(flag==2)
BitBlt(hdce,0,0,dx,dy,hdctmp1,x,y,SRCCOPY);
else
BitBlt(hdce,0,0,dx,dy,hdc,x,y,SRCCOPY);

size=bih.biSizeImage;

fwrite(&bfh,14,1,fpo);
fwrite(&bih,40,1,fpo);
fwrite(gdata,size,1,fpo);

fclose(fpo);

if(flag==3) DeleteDC(hdc);

```

```

DeleteDC(hdce);
DeleteObject(hbitmape);
}
else{
/* load */
if((fpi=fopen(file,"rb"))==NULL) {printf("Can't open the file.\n");return;}

fread(&bfh,14,1,fpi);
if(bfh.bfType!=0x4d42) {fclose(fpi);printf("Not BM.\n");return;}
fread(&bih,40,1,fpi);

fseek(fpi,bfh.bfOffBits,0);
size=bih.biSizeImage;
gdata=(BYTE *)malloc(size);
fread(gdata,size,1,fpi);

/*StretchDIBits(hdctmp2,x,y,bih.biWidth,bih.biHeight,0,0,bih.biWidth,bih.biHeight,
gdata,(LPBITMAPINFO)&bih,DIB_RGB_COLORS,SRCCOPY);*/

fclose(fpi);
free(gdata);
}
}/** ls_image **/

void fprintf_(char *str,int v2,int v3,int v4,int v5,int v6)
{
FILE *fp;

fp=fopen("cpage.bin","ab");

fprintf(fp," %s %d %d %d %d %d\n",str,v2,v3,v4,v5,v6);

fclose(fp);
}/** fprintf_ **/

void use_subroop(void)
{
char function_old,charflag_old;

usflag=1;

function_old=function;function=2;
charflag_old=charflag;

yorn=subroop();

```



```

function=function_old;
charflag=charflag_old;
}/** use_subroop **/

unsigned char subroop(void)
{
charflag=1;

while(1){
kbhit_();
if(charflag==0) return charcode;
}
}/** subroop **/

void keydowns_f2(void)
{
int dy;

if(GKS(VK_ESCAPE)<0 || GKS(VK_PAUSE)<0) charflag=0;
else if(GKS('S')<0){
dy=Y0+(yt+1/*1.7*/)*PIXSIZE+10;
ls_image(2,"ss.bmp",0,0,XRES0,/*YRES0*/dy);
/*printf(" %d\n",dy);*/
beep(300);
}
}/** keydowns_f2 **/

void restore_in_PAINT(void)
{
ValidateRect(hwnd,NULL);

bitblt(1,0,0,XRES0,YRES0,0,0);
}/** restore_in_PAINT **/

void setup(void)
{
XRES0=1024-4;YRES0=768-24*2;
}/** setup **/

int setup_(void)

```

```

{
int i,dy,n,m;

n=RES0;

/* a */
m=0;
i=5; /* a25 */
iv[m][i-5][0][0]=7*(n-1)-(N1-1); /* a->d */
iv[m][i-5][0][1]=3*(N2-1);
iv[m][i-5][1][0]=8*(n-1)-(N1-1); /* d->e */
iv[m][i-5][1][1]=0*(N2-1);
i=6; /* a26 */
iv[m][i-5][0][0]=8*(n-1)-(N1-1); /* a->e */
iv[m][i-5][0][1]=3*(N2-1);
iv[m][i-5][1][0]=5*(n-1)-(N1-1); /* e->b */
iv[m][i-5][1][1]=0*(N2-1);
i=7; /* a27 */
iv[m][i-5][0][0]=5*(n-1)-(N1-1); /* a->b */
iv[m][i-5][0][1]=3*(N2-1);
iv[m][i-5][1][0]=6*(n-1)-(N1-1); /* b->c */
iv[m][i-5][1][1]=0*(N2-1);
i=8; /* a28 */
iv[m][i-5][0][0]=6*(n-1)-(N1-1); /* a->c */
iv[m][i-5][0][1]=3*(N2-1);
iv[m][i-5][1][0]=7*(n-1)-(N1-1); /* a->d */
iv[m][i-5][1][1]=0*(N2-1);
i=9; /* a25 */
iv[m][i-5][0][0]=7*(n-1)-(N1-1); /* a->d */
iv[m][i-5][0][1]=3*(N2-1);
iv[m][i-5][1][0]=8*(n-1)-(N1-1); /* d->e */
iv[m][i-5][1][1]=0*(N2-1);

/* b */
m=1;
i=5; /* b26 */
iv[m][i-5][0][0]=4*(n-1)-(N1-1); /* b->a */
iv[m][i-5][0][1]=1*(N2-1);
iv[m][i-5][1][0]=8*(n-1)-(N1-1); /* a->e */
iv[m][i-5][1][1]=3*(N2-1);
i=6; /* b29 */
iv[m][i-5][0][0]=8*(n-1)-(N1-1); /* b->e */
iv[m][i-5][0][1]=2*(N2-1);
iv[m][i-5][1][0]=9*(n-1)-(N1-1); /* e->f */
iv[m][i-5][1][1]=0*(N2-1);
i=7; /* b32 */

```

```

iv[m][i-5][0][0]=9*(n-1)-(N1-1); /* b->f */
iv[m][i-5][0][1]=3*(N2-1);
iv[m][i-5][1][0]=6*(n-1)-(N1-1); /* f->c */
iv[m][i-5][1][1]=1*(N2-1);
i=8; /* b27 */
iv[m][i-5][0][0]=6*(n-1)-(N1-1); /* b->c */
iv[m][i-5][0][1]=0*(N2-1);
iv[m][i-5][1][0]=4*(n-1)-(N1-1); /* c->a */
iv[m][i-5][1][1]=2*(N2-1);
i=9; /* b26 */
iv[m][i-5][0][0]=4*(n-1)-(N1-1); /* b->a */
iv[m][i-5][0][1]=1*(N2-1);
iv[m][i-5][1][0]=8*(n-1)-(N1-1); /* a->e */
iv[m][i-5][1][1]=3*(N2-1);

```

```

/* c */

```

```

m=2;
i=5; /* c27 */
iv[m][i-5][0][0]=4*(n-1)-(N1-1); /* c->a */
iv[m][i-5][0][1]=2*(N2-1);
iv[m][i-5][1][0]=5*(n-1)-(N1-1); /* a->b */
iv[m][i-5][1][1]=3*(N2-1);
i=6; /* c32 */
iv[m][i-5][0][0]=5*(n-1)-(N1-1); /* c->b */
iv[m][i-5][0][1]=2*(N2-1);
iv[m][i-5][1][0]=9*(n-1)-(N1-1); /* b->f */
iv[m][i-5][1][1]=3*(N2-1);
i=7; /* c31 */
iv[m][i-5][0][0]=9*(n-1)-(N1-1); /* c->f */
iv[m][i-5][0][1]=2*(N2-1);
iv[m][i-5][1][0]=7*(n-1)-(N1-1); /* f->d */
iv[m][i-5][1][1]=1*(N2-1);
i=8; /* c28 */
iv[m][i-5][0][0]=7*(n-1)-(N1-1); /* c->d */
iv[m][i-5][0][1]=0*(N2-1);
iv[m][i-5][1][0]=4*(n-1)-(N1-1); /* d->a */
iv[m][i-5][1][1]=3*(N2-1);
i=9; /* c27 */
iv[m][i-5][0][0]=4*(n-1)-(N1-1); /* c->a */
iv[m][i-5][0][1]=2*(N2-1);
iv[m][i-5][1][0]=5*(n-1)-(N1-1); /* a->b */
iv[m][i-5][1][1]=3*(N2-1);

```

```

/* d */

```

```

m=3;
i=5; /* d28 */

```

```

iv[m][i-5][0][0]=4*(n-1)-(N1-1); /* d->a */
iv[m][i-5][0][1]=3*(N2-1);
iv[m][i-5][1][0]=6*(n-1)-(N1-1); /* a->c */
iv[m][i-5][1][1]=3*(N2-1);
i=6; /* d31 */
iv[m][i-5][0][0]=6*(n-1)-(N1-1); /* d->c */
iv[m][i-5][0][1]=2*(N2-1);
iv[m][i-5][1][0]=9*(n-1)-(N1-1); /* c->f */
iv[m][i-5][1][1]=2*(N2-1);
i=7; /* d30 */
iv[m][i-5][0][0]=9*(n-1)-(N1-1); /* d->f */
iv[m][i-5][0][1]=1*(N2-1);
iv[m][i-5][1][0]=8*(n-1)-(N1-1); /* f->e */
iv[m][i-5][1][1]=1*(N2-1);
i=8; /* d25 */
iv[m][i-5][0][0]=8*(n-1)-(N1-1); /* d->e */
iv[m][i-5][0][1]=0*(N2-1);
iv[m][i-5][1][0]=4*(n-1)-(N1-1); /* e->a */
iv[m][i-5][1][1]=0*(N2-1);
i=9; /* d28 */
iv[m][i-5][0][0]=4*(n-1)-(N1-1); /* d->a */
iv[m][i-5][0][1]=3*(N2-1);
iv[m][i-5][1][0]=6*(n-1)-(N1-1); /* a->c */
iv[m][i-5][1][1]=3*(N2-1);

/* e */
m=4;
i=5; /* e25 */
iv[m][i-5][0][0]=4*(n-1)-(N1-1); /* e->a */
iv[m][i-5][0][1]=0*(N2-1);
iv[m][i-5][1][0]=7*(n-1)-(N1-1); /* a->d */
iv[m][i-5][1][1]=3*(N2-1);
i=6; /* e30 */
iv[m][i-5][0][0]=7*(n-1)-(N1-1); /* e->d */
iv[m][i-5][0][1]=2*(N2-1);
iv[m][i-5][1][0]=9*(n-1)-(N1-1); /* d->f */
iv[m][i-5][1][1]=1*(N2-1);
i=7; /* e29 */
iv[m][i-5][0][0]=9*(n-1)-(N1-1); /* e->f */
iv[m][i-5][0][1]=0*(N2-1);
iv[m][i-5][1][0]=5*(n-1)-(N1-1); /* f->b */
iv[m][i-5][1][1]=1*(N2-1);
i=8; /* e26 */
iv[m][i-5][0][0]=5*(n-1)-(N1-1); /* e->b */
iv[m][i-5][0][1]=0*(N2-1);
iv[m][i-5][1][0]=4*(n-1)-(N1-1); /* b->a */

```

```

iv[m][i-5][1][1]=1*(N2-1);
i=9; /* e25 */
iv[m][i-5][0][0]=4*(n-1)-(N1-1); /* e->a */
iv[m][i-5][0][1]=0*(N2-1);
iv[m][i-5][1][0]=7*(n-1)-(N1-1); /* a->d */
iv[m][i-5][1][1]=3*(N2-1);

/* f */
m=5;
i=5; /* f29 */
iv[m][i-5][0][0]=5*(n-1)-(N1-1); /* f->b */
iv[m][i-5][0][1]=1*(N2-1);
iv[m][i-5][1][0]=8*(n-1)-(N1-1); /* b->e */
iv[m][i-5][1][1]=2*(N2-1);
i=6; /* f30 */
iv[m][i-5][0][0]=8*(n-1)-(N1-1); /* f->e */
iv[m][i-5][0][1]=1*(N2-1);
iv[m][i-5][1][0]=7*(n-1)-(N1-1); /* e->d */
iv[m][i-5][1][1]=2*(N2-1);
i=7; /* f31 */
iv[m][i-5][0][0]=7*(n-1)-(N1-1); /* f->d */
iv[m][i-5][0][1]=1*(N2-1);
iv[m][i-5][1][0]=6*(n-1)-(N1-1); /* d->c */
iv[m][i-5][1][1]=2*(N2-1);
i=8; /* f32 */
iv[m][i-5][0][0]=6*(n-1)-(N1-1); /* f->c */
iv[m][i-5][0][1]=1*(N2-1);
iv[m][i-5][1][0]=5*(n-1)-(N1-1); /* c->b */
iv[m][i-5][1][1]=2*(N2-1);
i=9; /* f29 */
iv[m][i-5][0][0]=5*(n-1)-(N1-1); /* f->b */
iv[m][i-5][0][1]=1*(N2-1);
iv[m][i-5][1][0]=8*(n-1)-(N1-1); /* b->e */
iv[m][i-5][1][1]=2*(N2-1);

PIXSIZE=10;
dy=Y0+(yt+1/*1.7*/)*PIXSIZE+10;

if(dy>dyMAX){
while(1){
PIXSIZE--;
dy=Y0+(yt+1/*1.7*/)*PIXSIZE+10;
if(dy<=dyMAX) break;
}
}

```

```

if(PIXSIZE<4) PIXSIZE=4;
if(1) PIXSIZE=7;

pixel=(*unsigned */char **)malloc(sizeof(*unsigned */char *)*((xt+1)+1));
if(pixel==NULL){
DeleteDC(hdctmp1);
DeleteObject(hbitmap1);
initgraph_return();return 1;}

i=0;
while(1){
pixel[i]=(*unsigned */char *)malloc(sizeof(*unsigned */char)*((yt+1)+1));

if(pixel[i]==NULL){
while(1){
i--;
if(i<0) break;
free(pixel[i]);
}
free(pixel);
DeleteDC(hdctmp1);
DeleteObject(hbitmap1);
initgraph_return();return 1;}

i++;
if(i==(xt+1)+1) break;
}

return 0;
}/** setup_ */

int initgraph_(void)
{
int i,width,height;
WNDCLASS wndclass;

setup();

wndclass.hInstance =hinstance;
wndclass.lpszClassName="CAGCLASS";
wndclass.lpszMenuName =NULL;
wndclass.lpfWndProc =wndproc_by_kbhit_;
wndclass.style =0;
wndclass.hIcon =LoadIcon(hinstance,"MYICON");
wndclass.hCursor =LoadCursor(NULL, IDC_ARROW);

```

```

wndclass.cbClsExtra    =0;
wndclass.cbWndExtra    =0;
if(WB==0)
wndclass.hbrBackground=GetStockObject(WHITE_BRUSH);
else
wndclass.hbrBackground=GetStockObject(BLACK_BRUSH);

if(RegisterClass(&wndclass)==0) return 1;

hwnd=CreateWindow("CAGCLASS"," CAG",
                /*WS_POPUP,*/
                WS_OVERLAPPED | WS_CAPTION | WS_SYSMENU | WS_MINIMIZEBOX,
                0+130,0,XRESO+DX_FRAME,YRESO+DY_CAPTION+DY_FRAME,
                NULL,NULL,hinstance,NULL);
if(hwnd==NULL) {MessageBox(NULL,"Memory space is not left.,"CAG",MB_OK);return 1;}

SetWindowPos(hwnd,HWND_TOP,0,0,0,0,SWP_NOMOVE | SWP_NOSIZE);
ShowWindow(hwnd,SW_SHOWDEFAULT);

hdcdisplay=GetDC(hwnd);

hbitmap1=CreateCompatibleBitmap(hdcdisplay,XRESO,YRESO);
hdctmp1=CreateCompatibleDC(hdcdisplay); /* text, dialog, menu */
SelectObject(hdctmp1,hbitmap1);
SetBkMode(hdcdisplay,TRANSPARENT);
SetBkMode(hdctmp1,TRANSPARENT);

initpalette();

SetBkColor(hdcdisplay,PALETTE(bfset[WB].back));
SetBkColor(hdctmp1,PALETTE(bfset[WB].back));

return 0;
}/** initgraph_ */

void initgraph_return(void)
{
/*EndPaint(hwnd,&paintstruct);*/
ReleaseDC(hwnd,hdcdisplay);
DestroyWindow(hwnd);
/*UnregisterClass("CAGCLASS",hinstance);*/

MessageBox(NULL,"Memory space is not left.,"CAG",MB_OK);
}/** initgraph_return */

```

```

void closegraph_(void)
{
int i;

i=0;
while(1){
free(pixel[i]);
i++;
if(i==(xt+1)+1) break;
}
free(pixel);

DeleteDC(hdctmp1);
DeleteObject(hbitmap1);

/*EndPoint(hwnd,&paintstruct);*/
ReleaseDC(hwnd,hdcdisplay);
DestroyWindow(hwnd);
/*UnregisterClass("CAGCLASS",hinstance);*/
}/** closegraph_ **/

void initpalette(void)
{
int i;

irgb[0].red=0;irgb[0].green=0;irgb[0].blue=0;

irgb[9].red=0;irgb[9].green=0;irgb[9].blue=255; /* blue */
irgb[10].red=0;irgb[10].green=255;irgb[10].blue=0; /* green */
irgb[11].red=0;irgb[11].green=255;irgb[11].blue=255; /* cyan */
irgb[12].red=255;irgb[12].green=0;irgb[12].blue=0; /* red */
irgb[13].red=255;irgb[13].green=0;irgb[13].blue=255; /* magenta */
irgb[14].red=255;irgb[14].green=255;irgb[14].blue=0; /* yellow */

irgb[15].red=255;irgb[15].green=255;irgb[15].blue=255;

for(i=7;i<9;i++){ /* 7, 8 */
irgb[i].red=128+32*(9-i);
irgb[i].green=irgb[i].red;
irgb[i].blue=irgb[i].red;
}

for(i=16;i<20;i++){ /* 16 -> 19 */
irgb[i].red=255-24*(20-i);

```



```
irgb[i].green=irgb[i].red;
irgb[i].blue=irgb[i].red;
}
```

```
for(i=1;i<7;i++){ /* 1 -> 6 */
if(irgb[9+(i-1)].red==255)
irgb[i].red=irgb[9+(i-1)].red-24*1;
if(irgb[9+(i-1)].green==255)
irgb[i].green=irgb[9+(i-1)].green-24*1;
if(irgb[9+(i-1)].blue==255)
irgb[i].blue=irgb[9+(i-1)].blue-24*1;
}
```

```
for(i=20;i<26;i++){ /* 20 -> 25 */
if(irgb[9+(i-20)].red==255)
irgb[i].red=irgb[9+(i-20)].red-24*2;
if(irgb[9+(i-20)].green==255)
irgb[i].green=irgb[9+(i-20)].green-24*2;
if(irgb[9+(i-20)].blue==255)
irgb[i].blue=irgb[9+(i-20)].blue-24*2;
}
```

```
for(i=26;i<32;i++){ /* 26 -> 31 */
if(irgb[9+(i-26)].red==255)
irgb[i].red=irgb[9+(i-26)].red-24*3;
if(irgb[9+(i-26)].green==255)
irgb[i].green=irgb[9+(i-26)].green-24*3;
if(irgb[9+(i-26)].blue==255)
irgb[i].blue=irgb[9+(i-26)].blue-24*3;
}
```

```
for(i=32;i<38;i++){ /* 32 -> 37 */
if(irgb[9+(i-32)].red==255)
irgb[i].red=irgb[9+(i-32)].red-24*4;
if(irgb[9+(i-32)].green==255)
irgb[i].green=irgb[9+(i-32)].green-24*4;
if(irgb[9+(i-32)].blue==255)
irgb[i].blue=irgb[9+(i-32)].blue-24*4;
}
```

```
for(i=38;i<44;i++){ /* 38 -> 43 */
if(irgb[9+(i-38)].red==255)
irgb[i].red=irgb[9+(i-38)].red-24*5;
if(irgb[9+(i-38)].green==255)
irgb[i].green=irgb[9+(i-38)].green-24*5;
if(irgb[9+(i-38)].blue==255)
```

```

irgb[i].blue=irgb[9+(i-38)].blue-24*5;
}

for(i=44;i<50;i++){
/* 44 -> 49 */
if(irgb[9+(i-44)].red==255)
irgb[i].red=irgb[9+(i-44)].red-24*6;
if(irgb[9+(i-44)].green==255)
irgb[i].green=irgb[9+(i-44)].green-24*6;
if(irgb[9+(i-44)].blue==255)
irgb[i].blue=irgb[9+(i-44)].blue-24*6;
}
}/** initpalette **/

void BitBlt_full(void)
{
bitblt(1,0,0,XRESO,YRESO,0,0);
}/** BitBlt_full **/

void bitblt(char flag,int x,int y,int xsize,int ysize,int x_,int y_)
{
BitBlt(hdcdisplay,x_,y_,xsize,ysize,
hdctmp1,x,y,SRCCOPY);
}/** bitblt **/

void cleardevice_(char flag,int x,int y,int xsize,int ysize)
{
PatBlt(hdctmp1,x,y,xsize,ysize,bfset[WB].back_);
}/** cleardevice_ **/

COLORREF PALETTE(int color)
{
return RGB(irgb[color].red,irgb[color].green,irgb[color].blue);
}/** PALETTE **/

void kbhit_(void)
{
MSG msg;

if(PeekMessage(&msg,NULL,0,0,PM_REMOVE)){
TranslateMessage(&msg);
DispatchMessage(&msg);
}
}

```

```

}
}/** kbhit_ */

LRESULT CALLBACK wndproc_by_kbhit_(HWND hwnd,UINT umsg,WPARAM wparam,LPARAM lparam)
{
if(wndproc_filer(hwnd,umsg,wparam,lparam)!=0) return 1;

return DefWindowProc(hwnd,umsg,wparam,lparam);
}/** wndproc_by_kbhit_ */

int wndproc_filer(HWND hwnd,UINT umsg,WPARAM wparam,LPARAM lparam)
{
if(umsg==WM_KEYDOWN){
/***** menu keydowns -> *****/
/***** <- menu keydowns *****/

/***** dialog keydowns -> *****/
/***** <- dialog keydowns *****/

if(function==2){
keydowns_f2();
return 1;
}

if(usflag==1) usflag=0;

if(GKS(VK_ESCAPE)<0 || GKS(VK_PAUSE)<0) refill=0;
else if(GKS(VK_SHIFT)<0) pauseflag=1;

return 1;
}/**else if(umsg)**/
else if(umsg==WM_SYSKEYDOWN){
}/**else if(umsg)**/
else if(umsg==WM_CLOSE){
if(function==2) charflag=0;
else refill=0;

return 1;
}/**else if(umsg)**/
else if(umsg==WM_PAINT){
restore_in_PAINT();

return 1;
}/**else if(umsg)**/

```

```

else{}

return 0;
}/** wndproc_filer **/

void delay_(long millisecond)
{
long oldtime,nowtime,dttime;
double i=CLOCKS_PER_SEC,j;

j=millisecond;
millisecond=j*(i/1000.);
oldtime=clock();

while(1){
kbhit_();
if(pauseflag==1 && refill==0) {pauseflag=0;refill=1;break;}
if(refill==0) break;

nowtime=clock();dttime=nowtime-oldtime;
if(dttime>=millisecond) break;
if(dttime<0) break;
}
}/** delay_ **/

void beep(long millisecond)
{
Beep(888,millisecond);
}/** beep **/

int fourfloor_fiveceil(double val_d)
{
int val_i,val;

val_i=floor(val_d);
val=(val_d-val_i<0.5)?val_i:val_i+1;

return val;
}/** fourfloor_fiveceil **/

int ff_fc(double val_d)
{

```

```

return fourfloor_fiveceil(val_d);
}/** ff_fc **/

void arrayreset(void)
{
int i,j;

i=0;
while(1){

j=0;
while(1){
pixel[i][j]=0/*1*/; /* here */
j++;
if(j==(yt+1)+1) break;
}

i++;
if(i==(xt+1)+1) break;
}
}/** arrayreset **/

int putpixel(int nx,int ny,int pcolor)
{
int i,dx,dy;
POINT vertex[7];

if(cnt<0) return 1;
if(nx<0 || ny<0) return 0;
if(N1<2) {if(nx>3*(RES0-1) || ny>4*(RES0-1)) return 0;}
else      {if(nx>xt+1 || ny>yt+1) return 0;}
if(GRPH==0) goto end;

dx=X0;
dy=Y0;
vertex[0].x=dx+nx*PIXSIZE;vertex[0].y=dy+ny*PIXSIZE;
vertex[1].x=dx+(nx+1)*PIXSIZE;vertex[1].y=dy+ny*PIXSIZE;
vertex[2].x=dx+(nx+1)*PIXSIZE;vertex[2].y=dy+(ny+1)*PIXSIZE;
vertex[3].x=dx+nx*PIXSIZE;vertex[3].y=dy+(ny+1)*PIXSIZE;
vertex[4].x=vertex[0].x;vertex[4].y=vertex[0].y;

if(pcolor==15)
hpen=CreatePen(PS_SOLID,1,PALETTE(9));
else

```

```

hpen=CreatePen(PS_SOLID,1,PALETTE(15));

/*if(nx==0 || ny==0 || nx==RESO-1 || ny==RESO-1) pcolor=12;*/
hbrush=CreateSolidBrush(PALETTE(pcolor));

SelectObject(hdcdisplay,hpen);
SelectObject(hdcdisplay,hbrush);

Polyline(hdcdisplay,vertex,/*6*/4+1);
Polygon(hdcdisplay,vertex,/*6*/4);

SelectObject(hdctmp1,hpen);
SelectObject(hdctmp1,hbrush);

Polyline(hdctmp1,vertex,/*6*/4+1);
Polygon(hdctmp1,vertex,/*6*/4);

DeleteObject(hbrush);
DeleteObject(hpen);

end:
pixel[nx][ny]=pcolor;

return 0;
}/** putpixel **/

void check_rcount(void)
{
int i;
long val[2];

if(GRPH>0){
for(i=0;i<CPMAX;i++)
printf(" %ld %ld\n",cnt,rcount[i]);
}
else if(1){
if(/*Odd>0 || CPMAX==5*/1){
val[0]=rcount[0];
for(i=1;i<CPMAX;i++){
if(rcount[i]!=val[0]) {beep(1000);refill=0;break;}
}

if(refill==0)
printf(" %ld %ld %d:%ld\n",cnt,val[0],i,rcount[i]);
else

```

```

printf(" %ld %ld\n",cnt,val[0]);
}/**if(_6dRow, CPMAX)**/
else{
val[0]=rcount[0];
for(i=1;i<CPHALF;i++){
if(rcount[i]!=val[0]) {beep(1000);refill=0;break;}
}

if(refill){
val[1]=rcount[CPHALF];
for(i=CPHALF+1;i<CPMAX;i++){
if(rcount[i]!=val[1]) {beep(1000);refill=0;break;}
}

if(refill)
printf(" %ld %ld %ld\n",cnt,val[0],val[1]);
else
printf(" %ld 1st:%ld 2nd:%ld %d:%ld\n",cnt,val[0],val[1],i,rcount[i]);
}/**if(refill)**/
else{
printf(" %ld 1st:%ld %d:%ld\n",cnt,val[0],i,rcount[i]);
}/**else(refill)**/
}/**else(_6dRow, CPMAX)**/
}
else{
printf(" %ld %ld %ld\n",cnt,rcount[0],rcount[1]);
if(rcount[0]!=rcount[1]) {beep(100);refill=0;}
}

if(GRPH==0 && cnt==GRPH_0_MAX) {beep(100);refill=0;}
}/** check_rcount **/

void field_s_(int x,int y,int dx,int dy,int pcolor)
{
int i,j;

if(pcolor==15){
for(j=y;j<y+dy;j++)
for(i=x;i<x+dx;i++)
putpixel(i,j,pcolor);
}
else{
for(j=y;j<y+dy;j++)
for(i=x;i<x+dx;i++)
pixel[i][j]=pcolor;
}
}

```

```

}
}/** field_s_ **/

void field_s(int x,int y)
{
int i,j,n;

n=RESO;

for(j=y;j<y+n;j++)
for(i=x;i<x+n;i++)
putpixel(i,j,15);
}/** field_s **/

void field(void)
{
int i,j,n,color;

fieldflag=1;

n=RESO;
color=/*0*/1;

/* left */
field_s(0,0);
if(N1>=2) field_s_(0+N1,0+N1,n-2*N1,n-2*N1,color);

/* centre */
i=1;
for(j=0;j<4;j++){
field_s(i*(n-1),j*(n-1));
if(N1>=2) field_s_(i*(n-1)+N1,j*(n-1)+N1,n-2*N1,n-2*N1,color);
}

/* right */
field_s(2*(n-1),0);
if(N1>=2) field_s_(2*(n-1)+N1,0+N1,n-2*N1,n-2*N1,color);

/* inner */
if(N1>=2){
for(i=0;i<6;i++)
field_s_((4+i)*(n-1)-(N1-1),0,N1,4*(N2-1)+1,15);
}

```



```

fieldflag=0;
}/** field **/

void pp_(int nx,int ny,int dx,int pcolor)
{
nx+=sn;
nx*=RES0-1;ny*=RES0-1;
nx+=dx;
putpixel(nx,ny,pcolor);
}/** pp_ **/

void pp(int qx,int qy,int qpos,int pcolor)
{
int n,nx,ny;

n=RES0;

if(qx<=3){
if(qpos==0) {nx=qx*(n-1)+N1-1;ny=qy*(n-1)+N1-1;}
else if(qpos==1) {nx=(qx+1)*(n-1)-(N1-1);ny=qy*(n-1)+N1-1;}
else if(qpos==2) {nx=(qx+1)*(n-1)-(N1-1);ny=(qy+1)*(n-1)-(N1-1);}
else {nx=qx*(n-1)+N1-1;ny=(qy+1)*(n-1)-(N1-1);}
}
else{
nx=qx*(n-1);ny=(qpos%5)*(N2-1);
if(qpos>4) nx-=N1-1;
}

putpixel(nx,ny,pcolor);
}/** pp **/

void rot(int pos,int dth)
{
pos=pos%4;
dth=dth%4;

pos+=dth;

if(pos>3) pos-=4;
else if(pos<0) pos+=4;

if(pos==0) {tmp0=1;tmp1=0;} /* ca1 */
else if(pos==1) {tmp0=0;tmp1=1;} /* ca2 */

```

```

else if(pos==2) {tmp0=-1;tmp1=0;} /* ca3 */
else if(pos==3) {tmp0=0;tmp1=-1;} /* ca4 */
}/** rot **/

```

```

int nh(int x,int y,int nx,int ny,int pos)
{
pos=pos%6;

if(pos==0){
if(nx==x+1 && ny==y) return 1;else return 0;
}
else if(pos==1){
if(nx==x && ny==y+1) return 1;else return 0;
}
else if(pos==2){
if(nx==x-1 && ny==y) return 1;else return 0;
}
else if(pos==3){
if(nx==x && ny==y-1) return 1;else return 0;
}
}/** nh **/

```

```

int side(int s,int side,int nx,int ny)
{
int n;

n=RES0;

if(s==/*1+0*/1 && side==0){
if(nx>=0*(n-1)+1 && nx<=1*(n-1)-1 && ny==/*3*/0*(n-1)) return 1;else return 0;
}
else if(s==/*1+3*/4 && side==3){
if(nx==1*(n-1) && ny>=2*(n-1)+1 && ny<=3*(n-1)-1) return 1;else return 0;
}
else if(s==/*1+0*/1 && side==3){
if(nx==0*(n-1) && ny>=/*3*/0*(n-1)+1 && ny<=/*4*/1*(n-1)-1) return 1;else return 0;
}
else if(s==/*1+2*/3 && side==3){
if(nx==1*(n-1) && ny>=1*(n-1)+1 && ny<=2*(n-1)-1) return 1;else return 0;
}
else if(s==/*1+0*/1 && side==2){
if(nx>=0*(n-1)+1 && nx<=1*(n-1)-1 && ny==/*4*/1*(n-1)) return 1;else return 0;
}
else if(s==5 && side==3){ /* ? */

```

```

if(nx==1*(n-1) && ny>=/*0*/3*(n-1)+1 && ny<=/*1*/4*(n-1)-1) return 1;else return 0;
}

else if(s==/*1+1*/2 && side==0){
if(nx>=1*(n-1)+1 && nx<=2*(n-1)-1 && ny==0*(n-1)) return 1;else return 0;
}
else if(s==/*1+4*/5 && side==2){
if(nx>=1*(n-1)+1 && nx<=2*(n-1)-1 && ny==4*(n-1)) return 1;else return 0;
}

else if(s==/*1+5*/6 && side==0){
if(nx>=2*(n-1)+1 && nx<=3*(n-1)-1 && ny==/*3*/0*(n-1)) return 1;else return 0;
}
else if(s==/*1+3*/4 && side==1){
if(nx==2*(n-1) && ny>=2*(n-1)+1 && ny<=3*(n-1)-1) return 1;else return 0;
}
else if(s==/*1+5*/6 && side==1){
if(nx==3*(n-1) && ny>=/*3*/0*(n-1)+1 && ny<=/*4*/1*(n-1)-1) return 1;else return 0;
}
else if(s==/*1+2*/3 && side==1){
if(nx==2*(n-1) && ny>=1*(n-1)+1 && ny<=2*(n-1)-1) return 1;else return 0;
}
else if(s==/*1+5*/6 && side==2){
if(nx>=2*(n-1)+1 && nx<=3*(n-1)-1 && ny==/*4*/1*(n-1)) return 1;else return 0;
}
else if(s==5 && side==1){ /* ? */
if(nx==2*(n-1) && ny>=/*0*/3*(n-1)+1 && ny<=/*1*/4*(n-1)-1) return 1;else return 0;
}

return 0;
}/** side **/

int side_(char s,int side,int nx,int ny)
{
int n,qx;

n=RES0;

/* outer */
if((qx=0)>-1 && s=='A' && side==0){
if(nx>=/*0*/qx*(n-1)+N1 && nx<=/*1*/(qx+1)*(n-1)-N1 && ny==0*(n-1)+N1-1) return 1;
else return 0;
}
else if(s=='A' && side==2){
if(nx>=/*0*/qx*(n-1)+N1 && nx<=/*1*/(qx+1)*(n-1)-N1 && ny==1*(n-1)-(N1-1)) return 1;

```

```

else return 0;
}
else if(s=='A' && side==3){
if(nx==/*0*/qx*(n-1)+(N1-1) && ny>=0*(n-1)+N1 && ny<=1*(n-1)-N1) return 1;
else return 0;
}
else if(s=='A' && side==1){
if(nx==/*1*/(qx+1)*(n-1)-(N1-1) && ny>=0*(n-1)+N1 && ny<=1*(n-1)-N1) return 1;
else return 0;
}

else if((qx=0)>-1 && s=='B' && side==0){
if(nx>=1*(n-1)+N1 && nx<=2*(n-1)-N1 && ny==qx*(n-1)+N1-1) return 1;
else return 0;
}
else if(s=='B' && side==2){
if(nx>=1*(n-1)+N1 && nx<=2*(n-1)-N1 && ny==(qx+1)*(n-1)-(N1-1)) return 1;
else return 0;
}
else if(s=='B' && side==3){
if(nx==n-1+N1-1 && ny>=qx*(n-1)+N1 && ny<=(qx+1)*(n-1)-N1) return 1;
else return 0;
}
else if(s=='B' && side==1){
if(nx==2*(n-1)-(N1-1) && ny>=qx*(n-1)+N1 && ny<=(qx+1)*(n-1)-N1) return 1;
else return 0;
}

else if((qx=1)>-1 && s=='C' && side==0){
if(nx>=1*(n-1)+N1 && nx<=2*(n-1)-N1 && ny==qx*(n-1)+N1-1) return 1;
else return 0;
}
else if(s=='C' && side==2){
if(nx>=1*(n-1)+N1 && nx<=2*(n-1)-N1 && ny==(qx+1)*(n-1)-(N1-1)) return 1;
else return 0;
}
else if(s=='C' && side==3){
if(nx==n-1+N1-1 && ny>=qx*(n-1)+N1 && ny<=(qx+1)*(n-1)-N1) return 1;
else return 0;
}
else if(s=='C' && side==1){
if(nx==2*(n-1)-(N1-1) && ny>=qx*(n-1)+N1 && ny<=(qx+1)*(n-1)-N1) return 1;
else return 0;
}

else if((qx=2)>-1 && s=='D' && side==0){

```

```

if(nx>=1*(n-1)+N1 && nx<=2*(n-1)-N1 && ny==qx*(n-1)+N1-1) return 1;
else return 0;
}
else if(s=='D' && side==2){
if(nx>=1*(n-1)+N1 && nx<=2*(n-1)-N1 && ny==(qx+1)*(n-1)-(N1-1)) return 1;
else return 0;
}
else if(s=='D' && side==3){
if(nx==n-1+N1-1 && ny>=qx*(n-1)+N1 && ny<=(qx+1)*(n-1)-N1) return 1;
else return 0;
}
else if(s=='D' && side==1){
if(nx==2*(n-1)-(N1-1) && ny>=qx*(n-1)+N1 && ny<=(qx+1)*(n-1)-N1) return 1;
else return 0;
}

else if((qx=3)>-1 && s=='E' && side==0){
if(nx>=1*(n-1)+N1 && nx<=2*(n-1)-N1 && ny==qx*(n-1)+N1-1) return 1;
else return 0;
}
else if(s=='E' && side==2){
if(nx>=1*(n-1)+N1 && nx<=2*(n-1)-N1 && ny==(qx+1)*(n-1)-(N1-1)) return 1;
else return 0;
}
else if(s=='E' && side==3){
if(nx==n-1+N1-1 && ny>=qx*(n-1)+N1 && ny<=(qx+1)*(n-1)-N1) return 1;
else return 0;
}
else if(s=='E' && side==1){
if(nx==2*(n-1)-(N1-1) && ny>=qx*(n-1)+N1 && ny<=(qx+1)*(n-1)-N1) return 1;
else return 0;
}

else if((qx=2)>-1 && s=='F' && side==0){
if(nx>=/*2*/qx*(n-1)+N1 && nx<=/*3*/(qx+1)*(n-1)-N1 && ny==0*(n-1)+N1-1) return 1;
else return 0;
}
else if(s=='F' && side==2){
if(nx>=/*2*/qx*(n-1)+N1 && nx<=/*3*/(qx+1)*(n-1)-N1 && ny==1*(n-1)-(N1-1)) return 1;
else return 0;
}
else if(s=='F' && side==3){
if(nx==/*2*/qx*(n-1)+N1-1 && ny>=0*(n-1)+N1 && ny<=1*(n-1)-N1) return 1;
else return 0;
}
else if(s=='F' && side==1){

```

```

if(nx==/*3*/(qx+1)*(n-1)-(N1-1) && ny>=0*(n-1)+N1 && ny<=1*(n-1)-N1) return 1;
else return 0;
}

```

```

/* inner(right) */

```

```

else if((qx=4)>-1 && s=='a' && side==/*1*/0){
if(nx==qx*(n-1) && ny>=0*(N2-1)+1 && ny<=1*(N2-1)-1) return 1;else return 0;
}

```

```

else if(s=='a' && side==/*3*/2){
if(nx==qx*(n-1) && ny>=2*(N2-1)+1 && ny<=3*(N2-1)-1) return 1;else return 0;
}

```

```

else if(s=='a' && side==/*4*/3){
if(nx==qx*(n-1) && ny>=3*(N2-1)+1 && ny<=4*(N2-1)-1) return 1;else return 0;
}

```

```

else if(s=='a' && side==/*2*/1){
if(nx==qx*(n-1) && ny>=1*(N2-1)+1 && ny<=2*(N2-1)-1) return 1;else return 0;
}

```

```

else if((qx=5)>-1 && s=='b' && side==/*1*/0){
if(nx==qx*(n-1) && ny>=0*(N2-1)+1 && ny<=1*(N2-1)-1) return 1;else return 0;
}

```

```

else if(s=='b' && side==/*3*/2){
if(nx==qx*(n-1) && ny>=2*(N2-1)+1 && ny<=3*(N2-1)-1) return 1;else return 0;
}

```

```

else if(s=='b' && side==/*4*/3){
if(nx==qx*(n-1) && ny>=3*(N2-1)+1 && ny<=4*(N2-1)-1) return 1;else return 0;
}

```

```

else if(s=='b' && side==/*2*/1){
if(nx==qx*(n-1) && ny>=1*(N2-1)+1 && ny<=2*(N2-1)-1) return 1;else return 0;
}

```

```

else if((qx=6)>-1 && s=='c' && side==/*1*/0){
if(nx==qx*(n-1) && ny>=0*(N2-1)+1 && ny<=1*(N2-1)-1) return 1;else return 0;
}

```

```

else if(s=='c' && side==/*3*/2){
if(nx==qx*(n-1) && ny>=2*(N2-1)+1 && ny<=3*(N2-1)-1) return 1;else return 0;
}

```

```

else if(s=='c' && side==/*4*/3){
if(nx==qx*(n-1) && ny>=3*(N2-1)+1 && ny<=4*(N2-1)-1) return 1;else return 0;
}

```

```

else if(s=='c' && side==/*2*/1){
if(nx==qx*(n-1) && ny>=1*(N2-1)+1 && ny<=2*(N2-1)-1) return 1;else return 0;
}

```

```

else if((qx=7)>-1 && s=='d' && side==/*1*/0){
if(nx==qx*(n-1) && ny>=0*(N2-1)+1 && ny<=1*(N2-1)-1) return 1;else return 0;
}

```

```

}
else if(s=='d' && side==/*3*/2){
if(nx==qx*(n-1) && ny>=2*(N2-1)+1 && ny<=3*(N2-1)-1) return 1;else return 0;
}
else if(s=='d' && side==/*4*/3){
if(nx==qx*(n-1) && ny>=3*(N2-1)+1 && ny<=4*(N2-1)-1) return 1;else return 0;
}
else if(s=='d' && side==/*2*/1){
if(nx==qx*(n-1) && ny>=1*(N2-1)+1 && ny<=2*(N2-1)-1) return 1;else return 0;
}

else if((qx=8)>-1 && s=='e' && side==/*1*/0){
if(nx==qx*(n-1) && ny>=0*(N2-1)+1 && ny<=1*(N2-1)-1) return 1;else return 0;
}
else if(s=='e' && side==/*3*/2){
if(nx==qx*(n-1) && ny>=2*(N2-1)+1 && ny<=3*(N2-1)-1) return 1;else return 0;
}
else if(s=='e' && side==/*4*/3){
if(nx==qx*(n-1) && ny>=3*(N2-1)+1 && ny<=4*(N2-1)-1) return 1;else return 0;
}
else if(s=='e' && side==/*2*/1){
if(nx==qx*(n-1) && ny>=1*(N2-1)+1 && ny<=2*(N2-1)-1) return 1;else return 0;
}

else if((qx=9)>-1 && s=='f' && side==/*1*/0){
if(nx==qx*(n-1) && ny>=0*(N2-1)+1 && ny<=1*(N2-1)-1) return 1;else return 0;
}
else if(s=='f' && side==/*3*/2){
if(nx==qx*(n-1) && ny>=2*(N2-1)+1 && ny<=3*(N2-1)-1) return 1;else return 0;
}
else if(s=='f' && side==/*4*/3){
if(nx==qx*(n-1) && ny>=3*(N2-1)+1 && ny<=4*(N2-1)-1) return 1;else return 0;
}
else if(s=='f' && side==/*2*/1){
if(nx==qx*(n-1) && ny>=1*(N2-1)+1 && ny<=2*(N2-1)-1) return 1;else return 0;
}

/* inner(centre) */
/* 9, 10 */
else if(s=='a' && side==/*0*/9){
if(nx>=4*(n-1)-(N1-1)+1 && nx<=4*(n-1)-1 && ny==0*(N2-1)) return 1;else return 0;
}
else if(s=='b' && side==/*0*/9){
if(nx>=5*(n-1)-(N1-1)+1 && nx<=5*(n-1)-1 && ny==0*(N2-1)) return 1;else return 0;
}
else if(s=='c' && side==/*0*/9){

```

```

if(nx>=6*(n-1)-(N1-1)+1 && nx<=6*(n-1)-1 && ny==0*(N2-1)) return 1;else return 0;
}
else if(s=='d' && side==/*0*/9){
if(nx>=7*(n-1)-(N1-1)+1 && nx<=7*(n-1)-1 && ny==0*(N2-1)) return 1;else return 0;
}
else if(s=='e' && side==/*0*/9){
if(nx>=8*(n-1)-(N1-1)+1 && nx<=8*(n-1)-1 && ny==0*(N2-1)) return 1;else return 0;
}
else if(s=='f' && side==/*0*/9){
if(nx>=9*(n-1)-(N1-1)+1 && nx<=9*(n-1)-1 && ny==0*(N2-1)) return 1;else return 0;
}
else if(s=='a' && side==/*5*/10){
if(nx>=4*(n-1)-(N1-1)+1 && nx<=4*(n-1)-1 && ny==4*(N2-1)) return 1;else return 0;
}
else if(s=='b' && side==/*5*/10){
if(nx>=5*(n-1)-(N1-1)+1 && nx<=5*(n-1)-1 && ny==4*(N2-1)) return 1;else return 0;
}
else if(s=='c' && side==/*5*/10){
if(nx>=6*(n-1)-(N1-1)+1 && nx<=6*(n-1)-1 && ny==4*(N2-1)) return 1;else return 0;
}
else if(s=='d' && side==/*5*/10){
if(nx>=7*(n-1)-(N1-1)+1 && nx<=7*(n-1)-1 && ny==4*(N2-1)) return 1;else return 0;
}
else if(s=='e' && side==/*5*/10){
if(nx>=8*(n-1)-(N1-1)+1 && nx<=8*(n-1)-1 && ny==4*(N2-1)) return 1;else return 0;
}
else if(s=='f' && side==/*5*/10){
if(nx>=9*(n-1)-(N1-1)+1 && nx<=9*(n-1)-1 && ny==4*(N2-1)) return 1;else return 0;
}

/* inner(left) */
/* A */
else if(s=='a' && side==/*9*/5){
if(nx==4*(n-1)-(N1-1) && ny>=0*(N2-1)+1 && ny<=1*(N2-1)-1) return 1;else return 0;
}
else if(s=='e' && side==/*6*/8){
if(nx==8*(n-1)-(N1-1) && ny>=3*(N2-1)+1 && ny<=4*(N2-1)-1) return 1;else return 0;
}
else if(s=='a' && side==7){
if(nx==4*(n-1)-(N1-1) && ny>=2*(N2-1)+1 && ny<=3*(N2-1)-1) return 1;else return 0;
}
else if(s=='c' && side==/*6*/8){
if(nx==6*(n-1)-(N1-1) && ny>=3*(N2-1)+1 && ny<=4*(N2-1)-1) return 1;else return 0;
}
else if(s=='a' && side==/*8*/6){
if(nx==4*(n-1)-(N1-1) && ny>=1*(N2-1)+1 && ny<=2*(N2-1)-1) return 1;else return 0;
}

```



```

}
else if(s=='b' && side==/*6*/8){
if(nx==5*(n-1)-(N1-1) && ny>=3*(N2-1)+1 && ny<=4*(N2-1)-1) return 1;else return 0;
}
else if(s=='a' && side==/*6*/8){
if(nx==4*(n-1)-(N1-1) && ny>=3*(N2-1)+1 && ny<=4*(N2-1)-1) return 1;else return 0;
}
else if(s=='d' && side==/*6*/8){
if(nx==7*(n-1)-(N1-1) && ny>=3*(N2-1)+1 && ny<=4*(N2-1)-1) return 1;else return 0;
}

/* F */
else if(s=='b' && side==/*8*/6){
if(nx==5*(n-1)-(N1-1) && ny>=1*(N2-1)+1 && ny<=2*(N2-1)-1) return 1;else return 0;
}
else if(s=='f' && side==/*6*/8){
if(nx==9*(n-1)-(N1-1) && ny>=3*(N2-1)+1 && ny<=4*(N2-1)-1) return 1;else return 0;
}
else if(s=='d' && side==/*8*/6){
if(nx==7*(n-1)-(N1-1) && ny>=1*(N2-1)+1 && ny<=2*(N2-1)-1) return 1;else return 0;
}
else if(s=='f' && side==/*8*/6){
if(nx==9*(n-1)-(N1-1) && ny>=1*(N2-1)+1 && ny<=2*(N2-1)-1) return 1;else return 0;
}
else if(s=='e' && side==/*8*/6){
if(nx==8*(n-1)-(N1-1) && ny>=1*(N2-1)+1 && ny<=2*(N2-1)-1) return 1;else return 0;
}
else if(s=='f' && side==/*9*/5){
if(nx==9*(n-1)-(N1-1) && ny>=0*(N2-1)+1 && ny<=1*(N2-1)-1) return 1;else return 0;
}
else if(s=='c' && side==/*8*/6){
if(nx==6*(n-1)-(N1-1) && ny>=1*(N2-1)+1 && ny<=2*(N2-1)-1) return 1;else return 0;
}
else if(s=='f' && side==7){
if(nx==9*(n-1)-(N1-1) && ny>=2*(N2-1)+1 && ny<=3*(N2-1)-1) return 1;else return 0;
}

/* B^E */
else if(s=='b' && side==/*9*/5){
if(nx==5*(n-1)-(N1-1) && ny>=0*(N2-1)+1 && ny<=1*(N2-1)-1) return 1;else return 0;
}
else if(s=='e' && side==7){
if(nx==8*(n-1)-(N1-1) && ny>=2*(N2-1)+1 && ny<=3*(N2-1)-1) return 1;else return 0;
}
else if(s=='b' && side==7){
if(nx==5*(n-1)-(N1-1) && ny>=2*(N2-1)+1 && ny<=3*(N2-1)-1) return 1;else return 0;
}

```

```

}
else if(s=='c' && side==/*9*/5){
if(nx==6*(n-1)-(N1-1) && ny>=0*(N2-1)+1 && ny<=1*(N2-1)-1) return 1;else return 0;
}
else if(s=='c' && side==7){
if(nx==6*(n-1)-(N1-1) && ny>=2*(N2-1)+1 && ny<=3*(N2-1)-1) return 1;else return 0;
}
else if(s=='d' && side==/*9*/5){
if(nx==7*(n-1)-(N1-1) && ny>=0*(N2-1)+1 && ny<=1*(N2-1)-1) return 1;else return 0;
}
else if(s=='d' && side==7){
if(nx==7*(n-1)-(N1-1) && ny>=2*(N2-1)+1 && ny<=3*(N2-1)-1) return 1;else return 0;
}
else if(s=='e' && side==/*9*/5){
if(nx==8*(n-1)-(N1-1) && ny>=0*(N2-1)+1 && ny<=1*(N2-1)-1) return 1;else return 0;
}

return 0;
}/** side_ **/

```

```

int v(int s,int v,int nx,int ny)
{
int n;

n=RES0;

if(s==/*1+0*/1 && v==/*0*/3){
if(nx==0*(n-1) && ny==/*3*/1*(n-1)) return 1;else return 0;
}
else if(s==/*1+3*/4 && v==0){
if(nx==1*(n-1) && ny==2*(n-1)) return 1;else return 0;
}
else if(s==/*1+0*/1 && v==/*2*/1){ /* ? */
if(nx==1*(n-1) && ny==/*4*/0*(n-1)) return 1;else return 0;
}
else if(s==5 && v==/*0*/3){
if(nx==1*(n-1) && ny==/*0*/4*(n-1)) return 1;else return 0;
}
else if(s==/*1+0*/1 && v==/*3*/0){
if(nx==0*(n-1) && ny==/*4*/0*(n-1)) return 1;else return 0;
}
else if(s==5 && v==0){
if(nx==1*(n-1) && ny==/*1*/3*(n-1)) return 1;else return 0;
}
}

```

```

else if(s==/*1+5*/6 && v==/*1*/2){
if(nx==3*(n-1) && ny==/*3*/1*(n-1)) return 1;else return 0;
}
else if(s==/*1+3*/4 && v==1){
if(nx==2*(n-1) && ny==2*(n-1)) return 1;else return 0;
}
else if(s==/*1+5*/6 && v==/*2*/1){
if(nx==3*(n-1) && ny==/*4*/0*(n-1)) return 1;else return 0;
}
else if(s==5 && v==1){
if(nx==2*(n-1) && ny==/*1*/3*(n-1)) return 1;else return 0;
}
else if(s==/*1+5*/6 && v==/*3*/0){ /* ? */
if(nx==2*(n-1) && ny==/*4*/0*(n-1)) return 1;else return 0;
}
else if(s==5 && v==/*1*/2){
if(nx==2*(n-1) && ny==/*0*/4*(n-1)) return 1;else return 0;
}

return 0;
}/** v **/

```

```

int v_(char s,int v,int nx,int ny)
{
int n,qx,qy,dlt;

n=RES0;
dlt=N1-1;

/* outer */
if((qx=0)>-1 && (qy=0)>-1 && s=='A' && v==0){
if(nx==qx*(n-1)+N1-1 && ny==qy*(n-1)+N1-1) return 1;else return 0;
}
else if(s=='A' && v==1){
if(nx==(qx+1)*(n-1)-(N1-1) && ny==qy*(n-1)+N1-1) return 1;else return 0;
}
else if(s=='A' && v==2){
if(nx==(qx+1)*(n-1)-(N1-1) && ny==(qy+1)*(n-1)-(N1-1)) return 1;else return 0;
}
else if(s=='A' && v==3){
if(nx==qx*(n-1)+N1-1 && ny==(qy+1)*(n-1)-(N1-1)) return 1;else return 0;
}

else if((qx=1)>-1 && (qy=0)>-1 && s=='B' && v==0){
if(nx==qx*(n-1)+N1-1 && ny==qy*(n-1)+N1-1) return 1;else return 0;
}

```

```

}
else if(s=='B' && v==1){
if(nx==(qx+1)*(n-1)-(N1-1) && ny==qy*(n-1)+N1-1) return 1;else return 0;
}
else if(s=='B' && v==2){
if(nx==(qx+1)*(n-1)-(N1-1) && ny==(qy+1)*(n-1)-(N1-1)) return 1;else return 0;
}
else if(s=='B' && v==3){
if(nx==qx*(n-1)+N1-1 && ny==(qy+1)*(n-1)-(N1-1)) return 1;else return 0;
}

else if((qx=1)>-1 && (qy=1)>-1 && s=='C' && v==0){
if(nx==qx*(n-1)+N1-1 && ny==qy*(n-1)+N1-1) return 1;else return 0;
}
else if(s=='C' && v==1){
if(nx==(qx+1)*(n-1)-(N1-1) && ny==qy*(n-1)+N1-1) return 1;else return 0;
}
else if(s=='C' && v==2){
if(nx==(qx+1)*(n-1)-(N1-1) && ny==(qy+1)*(n-1)-(N1-1)) return 1;else return 0;
}
else if(s=='C' && v==3){
if(nx==qx*(n-1)+N1-1 && ny==(qy+1)*(n-1)-(N1-1)) return 1;else return 0;
}

else if((qx=1)>-1 && (qy=2)>-1 && s=='D' && v==0){
if(nx==qx*(n-1)+N1-1 && ny==qy*(n-1)+N1-1) return 1;else return 0;
}
else if(s=='D' && v==1){
if(nx==(qx+1)*(n-1)-(N1-1) && ny==qy*(n-1)+N1-1) return 1;else return 0;
}
else if(s=='D' && v==2){
if(nx==(qx+1)*(n-1)-(N1-1) && ny==(qy+1)*(n-1)-(N1-1)) return 1;else return 0;
}
else if(s=='D' && v==3){
if(nx==qx*(n-1)+N1-1 && ny==(qy+1)*(n-1)-(N1-1)) return 1;else return 0;
}

else if((qx=1)>-1 && (qy=3)>-1 && s=='E' && v==0){
if(nx==qx*(n-1)+N1-1 && ny==qy*(n-1)+N1-1) return 1;else return 0;
}
else if(s=='E' && v==1){
if(nx==(qx+1)*(n-1)-(N1-1) && ny==qy*(n-1)+N1-1) return 1;else return 0;
}
else if(s=='E' && v==2){
if(nx==(qx+1)*(n-1)-(N1-1) && ny==(qy+1)*(n-1)-(N1-1)) return 1;else return 0;
}

```

```

else if(s=='E' && v==3){
if(nx==qx*(n-1)+N1-1 && ny==(qy+1)*(n-1)-(N1-1)) return 1;else return 0;
}

else if((qx=2)>-1 && (qy=0)>-1 && s=='F' && v==0){
if(nx==qx*(n-1)+N1-1 && ny==qy*(n-1)+N1-1) return 1;else return 0;
}
else if(s=='F' && v==1){
if(nx==(qx+1)*(n-1)-(N1-1) && ny==qy*(n-1)+N1-1) return 1;else return 0;
}
else if(s=='F' && v==2){
if(nx==(qx+1)*(n-1)-(N1-1) && ny==(qy+1)*(n-1)-(N1-1)) return 1;else return 0;
}
else if(s=='F' && v==3){
if(nx==qx*(n-1)+N1-1 && ny==(qy+1)*(n-1)-(N1-1)) return 1;else return 0;
}

/* inner(right) */
else if((qx=4)>-1 && s=='a' && v==0){
if(nx==qx*(n-1) && ny==0) return 1;else return 0;
}
else if(s=='a' && v==1){
if(nx==qx*(n-1) && ny==1*(N2-1)) return 1;else return 0;
}
else if(s=='a' && v==2){
if(nx==qx*(n-1) && ny==2*(N2-1)) return 1;else return 0;
}
else if(s=='a' && v==3){
if(nx==qx*(n-1) && ny==3*(N2-1)) return 1;else return 0;
}
else if(s=='a' && v==4){
if(nx==qx*(n-1) && ny==4*(N2-1)) return 1;else return 0;
}

else if((qx=5)>-1 && s=='b' && v==0){
if(nx==qx*(n-1) && ny==0) return 1;else return 0;
}
else if(s=='b' && v==1){
if(nx==qx*(n-1) && ny==1*(N2-1)) return 1;else return 0;
}
else if(s=='b' && v==2){
if(nx==qx*(n-1) && ny==2*(N2-1)) return 1;else return 0;
}
else if(s=='b' && v==3){
if(nx==qx*(n-1) && ny==3*(N2-1)) return 1;else return 0;
}

```

```
else if(s=='b' && v==4){
if(nx==qx*(n-1) && ny==4*(N2-1)) return 1;else return 0;
}
```

```
else if((qx=6)>-1 && s=='c' && v==0){
if(nx==qx*(n-1) && ny==0) return 1;else return 0;
}
else if(s=='c' && v==1){
if(nx==qx*(n-1) && ny==1*(N2-1)) return 1;else return 0;
}
else if(s=='c' && v==2){
if(nx==qx*(n-1) && ny==2*(N2-1)) return 1;else return 0;
}
else if(s=='c' && v==3){
if(nx==qx*(n-1) && ny==3*(N2-1)) return 1;else return 0;
}
else if(s=='c' && v==4){
if(nx==qx*(n-1) && ny==4*(N2-1)) return 1;else return 0;
}
```

```
else if((qx=7)>-1 && s=='d' && v==0){
if(nx==qx*(n-1) && ny==0) return 1;else return 0;
}
else if(s=='d' && v==1){
if(nx==qx*(n-1) && ny==1*(N2-1)) return 1;else return 0;
}
else if(s=='d' && v==2){
if(nx==qx*(n-1) && ny==2*(N2-1)) return 1;else return 0;
}
else if(s=='d' && v==3){
if(nx==qx*(n-1) && ny==3*(N2-1)) return 1;else return 0;
}
else if(s=='d' && v==4){
if(nx==qx*(n-1) && ny==4*(N2-1)) return 1;else return 0;
}
```

```
else if((qx=8)>-1 && s=='e' && v==0){
if(nx==qx*(n-1) && ny==0) return 1;else return 0;
}
else if(s=='e' && v==1){
if(nx==qx*(n-1) && ny==1*(N2-1)) return 1;else return 0;
}
else if(s=='e' && v==2){
if(nx==qx*(n-1) && ny==2*(N2-1)) return 1;else return 0;
}
else if(s=='e' && v==3){
```

```

if(nx==qx*(n-1) && ny==3*(N2-1)) return 1;else return 0;
}
else if(s=='e' && v==4){
if(nx==qx*(n-1) && ny==4*(N2-1)) return 1;else return 0;
}

else if((qx=9)>-1 && s=='f' && v==0){
if(nx==qx*(n-1) && ny==0) return 1;else return 0;
}
else if(s=='f' && v==1){
if(nx==qx*(n-1) && ny==1*(N2-1)) return 1;else return 0;
}
else if(s=='f' && v==2){
if(nx==qx*(n-1) && ny==2*(N2-1)) return 1;else return 0;
}
else if(s=='f' && v==3){
if(nx==qx*(n-1) && ny==3*(N2-1)) return 1;else return 0;
}
else if(s=='f' && v==4){
if(nx==qx*(n-1) && ny==4*(N2-1)) return 1;else return 0;
}

/* inner(left) */
else if((qx=4)>-1 && s=='a' && v==5){
if(nx==qx*(n-1)-dlt && ny==0) return 1;else return 0;
}
else if(s=='a' && v==6){
if(nx==qx*(n-1)-dlt && ny==1*(N2-1)) return 1;else return 0;
}
else if(s=='a' && v==7){
if(nx==qx*(n-1)-dlt && ny==2*(N2-1)) return 1;else return 0;
}
else if(s=='a' && v==8){
if(nx==qx*(n-1)-dlt && ny==3*(N2-1)) return 1;else return 0;
}
else if(s=='a' && v==9){
if(nx==qx*(n-1)-dlt && ny==4*(N2-1)) return 1;else return 0;
}

else if((qx=5)>-1 && s=='b' && v==5){
if(nx==qx*(n-1)-dlt && ny==0) return 1;else return 0;
}
else if(s=='b' && v==6){
if(nx==qx*(n-1)-dlt && ny==1*(N2-1)) return 1;else return 0;
}
else if(s=='b' && v==7){

```

```

if(nx==qx*(n-1)-dlt && ny==2*(N2-1)) return 1;else return 0;
}
else if(s=='b' && v==8){
if(nx==qx*(n-1)-dlt && ny==3*(N2-1)) return 1;else return 0;
}
else if(s=='b' && v==9){
if(nx==qx*(n-1)-dlt && ny==4*(N2-1)) return 1;else return 0;
}

else if((qx=6)>-1 && s=='c' && v==5){
if(nx==qx*(n-1)-dlt && ny==0) return 1;else return 0;
}
else if(s=='c' && v==6){
if(nx==qx*(n-1)-dlt && ny==1*(N2-1)) return 1;else return 0;
}
else if(s=='c' && v==7){
if(nx==qx*(n-1)-dlt && ny==2*(N2-1)) return 1;else return 0;
}
else if(s=='c' && v==8){
if(nx==qx*(n-1)-dlt && ny==3*(N2-1)) return 1;else return 0;
}
else if(s=='c' && v==9){
if(nx==qx*(n-1)-dlt && ny==4*(N2-1)) return 1;else return 0;
}

else if((qx=7)>-1 && s=='d' && v==5){
if(nx==qx*(n-1)-dlt && ny==0) return 1;else return 0;
}
else if(s=='d' && v==6){
if(nx==qx*(n-1)-dlt && ny==1*(N2-1)) return 1;else return 0;
}
else if(s=='d' && v==7){
if(nx==qx*(n-1)-dlt && ny==2*(N2-1)) return 1;else return 0;
}
else if(s=='d' && v==8){
if(nx==qx*(n-1)-dlt && ny==3*(N2-1)) return 1;else return 0;
}
else if(s=='d' && v==9){
if(nx==qx*(n-1)-dlt && ny==4*(N2-1)) return 1;else return 0;
}

else if((qx=8)>-1 && s=='e' && v==5){
if(nx==qx*(n-1)-dlt && ny==0) return 1;else return 0;
}
else if(s=='e' && v==6){
if(nx==qx*(n-1)-dlt && ny==1*(N2-1)) return 1;else return 0;
}

```



```

}
else if(s=='e' && v==7){
if(nx==qx*(n-1)-dlt && ny==2*(N2-1)) return 1;else return 0;
}
else if(s=='e' && v==8){
if(nx==qx*(n-1)-dlt && ny==3*(N2-1)) return 1;else return 0;
}
else if(s=='e' && v==9){
if(nx==qx*(n-1)-dlt && ny==4*(N2-1)) return 1;else return 0;
}

else if((qx=9)>-1 && s=='f' && v==5){
if(nx==qx*(n-1)-dlt && ny==0) return 1;else return 0;
}
else if(s=='f' && v==6){
if(nx==qx*(n-1)-dlt && ny==1*(N2-1)) return 1;else return 0;
}
else if(s=='f' && v==7){
if(nx==qx*(n-1)-dlt && ny==2*(N2-1)) return 1;else return 0;
}
else if(s=='f' && v==8){
if(nx==qx*(n-1)-dlt && ny==3*(N2-1)) return 1;else return 0;
}
else if(s=='f' && v==9){
if(nx==qx*(n-1)-dlt && ny==4*(N2-1)) return 1;else return 0;
}

return 0;
}/** v_ **/

```

```

void putpixel_(int nx,int ny,int pcolor)
{
int n,qx,qy,dlt,flag,snold;

n=RES0;

putpixel(nx,ny,pcolor);
if(fieldflag) return;
rcount[ig]++;

/*return;*/ /* here */
nx-=sn*(n-1);

if(v(*1+0*/1,*0*/3,nx,ny)==1 || v(*1+3*/4,0,nx,ny)==1){
pp_(0,*3*/1,0,pcolor);pp_(1,2,0,pcolor);

```

```

}
else if(v(/*1+0*/1,/*3*/0,nx,ny)==1 || v(/*1+2*/5,0,nx,ny)==1){
pp_(0,/*4*/0,0,pcolor);pp_(1,/*1*/3,0,pcolor);
}
else if(v(/*1+0*/1,/*2*/1,nx,ny)==1 || v(/*1+1*/5,/*0*/3,nx,ny)==1){
pp_(1,/*4*/0,0,pcolor);pp_(1,/*0*/4,0,pcolor);
}
else if(v(/*1+5*/6,/*1*/2,nx,ny)==1 || v(/*1+3*/4,1,nx,ny)==1){
pp_(3,/*3*/1,0,pcolor);pp_(2,/*2*/2,0,pcolor);
}
else if(v(/*1+5*/6,/*2*/1,nx,ny)==1 || v(/*1+2*/5,1,nx,ny)==1){
pp_(3,/*4*/0,0,pcolor);pp_(2,/*1*/3,0,pcolor);
}
else if(v(/*1+5*/6,/*3*/0,nx,ny)==1 || v(/*1+1*/5,/*1*/2,nx,ny)==1){
pp_(2,/*4*/0,0,pcolor);pp_(2,/*0*/4,0,pcolor);
}

else if((qx=0)>-1 && (qy=0)>-1 &&
(v_('A',0,nx,ny)==1 || v_('a',0,nx,ny)==1 || v_('a',4,nx,ny)==1)){
pp(qx,qy,0,pcolor); pp(qx+qy+4,0,0,pcolor);pp(qx+qy+4,0,4,pcolor);
}
else if(v_('A',1,nx,ny)==1 || v_('a',1,nx,ny)==1){
pp(qx,qy,1,pcolor); pp(qx+qy+4,0,1,pcolor);
}
else if(v_('A',2,nx,ny)==1 || v_('a',2,nx,ny)==1){
pp(qx,qy,2,pcolor); pp(qx+qy+4,0,2,pcolor);
}
else if(v_('A',3,nx,ny)==1 || v_('a',3,nx,ny)==1){
pp(qx,qy,3,pcolor); pp(qx+qy+4,0,3,pcolor);
}

else if((qx=1)>-1 && (qy=0)>-1 &&
v_('B',0,nx,ny)==1 || v_('b',0,nx,ny)==1 || v_('b',4,nx,ny)==1){
pp(qx,qy,0,pcolor); pp(qx+qy+4,0,0,pcolor);pp(qx+qy+4,0,4,pcolor);
}
else if(v_('B',1,nx,ny)==1 || v_('b',1,nx,ny)==1){
pp(qx,qy,1,pcolor); pp(qx+qy+4,0,1,pcolor);
}
else if(v_('B',2,nx,ny)==1 || v_('b',2,nx,ny)==1){
pp(qx,qy,2,pcolor); pp(qx+qy+4,0,2,pcolor);
}
else if(v_('B',3,nx,ny)==1 || v_('b',3,nx,ny)==1){
pp(qx,qy,3,pcolor); pp(qx+qy+4,0,3,pcolor);
}

else if((qx=1)>-1 && (qy=1)>-1 &&

```

```

        v_('C',0,nx,ny)==1 || v_('c',0,nx,ny)==1 || v_('c',4,nx,ny)==1){
pp(qx,qy,0,pcolor);          pp(qx+qy+4,0,0,pcolor);pp(qx+qy+4,0,4,pcolor);
}
else if(v_('C',1,nx,ny)==1 || v_('c',1,nx,ny)==1){
pp(qx,qy,1,pcolor);          pp(qx+qy+4,0,1,pcolor);
}
else if(v_('C',2,nx,ny)==1 || v_('c',2,nx,ny)==1){
pp(qx,qy,2,pcolor);          pp(qx+qy+4,0,2,pcolor);
}
else if(v_('C',3,nx,ny)==1 || v_('c',3,nx,ny)==1){
pp(qx,qy,3,pcolor);          pp(qx+qy+4,0,3,pcolor);
}

else if((qx=1)>-1 && (qy=2)>-1 &&
        v_('D',0,nx,ny)==1 || v_('d',0,nx,ny)==1 || v_('d',4,nx,ny)==1){
pp(qx,qy,0,pcolor);          pp(qx+qy+4,0,0,pcolor);pp(qx+qy+4,0,4,pcolor);
}
else if(v_('D',1,nx,ny)==1 || v_('d',1,nx,ny)==1){
pp(qx,qy,1,pcolor);          pp(qx+qy+4,0,1,pcolor);
}
else if(v_('D',2,nx,ny)==1 || v_('d',2,nx,ny)==1){
pp(qx,qy,2,pcolor);          pp(qx+qy+4,0,2,pcolor);
}
else if(v_('D',3,nx,ny)==1 || v_('d',3,nx,ny)==1){
pp(qx,qy,3,pcolor);          pp(qx+qy+4,0,3,pcolor);
}

else if((qx=1)>-1 && (qy=3)>-1 &&
        v_('E',0,nx,ny)==1 || v_('e',0,nx,ny)==1 || v_('e',4,nx,ny)==1){
pp(qx,qy,0,pcolor);          pp(qx+qy+4,0,0,pcolor);pp(qx+qy+4,0,4,pcolor);
}
else if(v_('E',1,nx,ny)==1 || v_('e',1,nx,ny)==1){
pp(qx,qy,1,pcolor);          pp(qx+qy+4,0,1,pcolor);
}
else if(v_('E',2,nx,ny)==1 || v_('e',2,nx,ny)==1){
pp(qx,qy,2,pcolor);          pp(qx+qy+4,0,2,pcolor);
}
else if(v_('E',3,nx,ny)==1 || v_('e',3,nx,ny)==1){
pp(qx,qy,3,pcolor);          pp(qx+qy+4,0,3,pcolor);
}

else if((qx=2)>-1 && (qy=0)>-1 &&
        v_('F',0,nx,ny)==1 || v_('f',0,nx,ny)==1 || v_('f',4,nx,ny)==1){
pp(qx,qy,0,pcolor);          pp(qx+3+4,0,0,pcolor);pp(qx+3+4,0,4,pcolor);
}
else if(v_('F',1,nx,ny)==1 || v_('f',1,nx,ny)==1){

```

```

pp(qx,qy,1,pcolor);          pp(qx+3+4,0,1,pcolor);
}
else if(v_('F',2,nx,ny)==1 || v_('f',2,nx,ny)==1){
pp(qx,qy,2,pcolor);          pp(qx+3+4,0,2,pcolor);
}
else if(v_('F',3,nx,ny)==1 || v_('f',3,nx,ny)==1){
pp(qx,qy,3,pcolor);          pp(qx+3+4,0,3,pcolor);
}

else if(v_('a',5,nx,ny)==1 || v_('d',8,nx,ny)==1 || v_('e',5,nx,ny)==1 ||
        v_('a',9,nx,ny)==1 || v_('e',9,nx,ny)==1){
pp(4,0,5,pcolor);pp(7,0,8,pcolor);pp(8,0,5,pcolor);pp(4,0,9,pcolor);pp(8,0,9,pcolor);
}
else if(v_('a',6,nx,ny)==1 || v_('b',5,nx,ny)==1 || v_('e',8,nx,ny)==1 ||
        v_('b',9,nx,ny)==1){
pp(4,0,6,pcolor);pp(5,0,5,pcolor);pp(8,0,8,pcolor);pp(5,0,9,pcolor);
}
else if(v_('a',7,nx,ny)==1 || v_('b',8,nx,ny)==1 || v_('c',5,nx,ny)==1 ||
        v_('c',9,nx,ny)==1){
pp(4,0,7,pcolor);pp(5,0,8,pcolor);pp(6,0,5,pcolor);pp(6,0,9,pcolor);
}
else if(v_('a',8,nx,ny)==1 || v_('c',8,nx,ny)==1 || v_('d',5,nx,ny)==1 ||
        v_('d',9,nx,ny)==1){
pp(4,0,8,pcolor);pp(6,0,8,pcolor);pp(7,0,5,pcolor);pp(7,0,9,pcolor);
}

else if(v_('f',5,nx,ny)==1 || v_('b',6,nx,ny)==1 || v_('e',7,nx,ny)==1 ||
        v_('f',9,nx,ny)==1){
pp(9,0,5,pcolor);pp(5,0,6,pcolor);pp(8,0,7,pcolor);pp(9,0,9,pcolor);
}
else if(v_('f',6,nx,ny)==1 || v_('d',7,nx,ny)==1 || v_('e',6,nx,ny)==1){
pp(9,0,6,pcolor);pp(7,0,7,pcolor);pp(8,0,6,pcolor);
}
else if(v_('f',7,nx,ny)==1 || v_('c',7,nx,ny)==1 || v_('d',6,nx,ny)==1){
pp(9,0,7,pcolor);pp(6,0,7,pcolor);pp(7,0,6,pcolor);
}
else if(v_('f',8,nx,ny)==1 || v_('b',7,nx,ny)==1 || v_('c',6,nx,ny)==1){
pp(9,0,8,pcolor);pp(5,0,7,pcolor);pp(6,0,6,pcolor);
}

else{
if(side(/*1+0*/1,/*0*/2,nx,ny)==1 && sn<8) flag=101;
else if(side(/*1+3*/3,3,nx,ny)==1 && sn<8) flag=102;
else if(side(/*1+0*/1,3,nx,ny)==1 && sn<8) flag=103;
else if(side(/*1+2*/4,3,nx,ny)==1 && sn<8) flag=104;
else if(side(/*1+0*/1,/*2*/0,nx,ny)==1 && sn<8) flag=105;
}

```

```

else if(side(/*1+1*/5,3,nx,ny)==1 && sn<8) flag=106;

else if(side(/*1+1*/5,/*0*/2,nx,ny)==1 && sn<8) flag=107;
else if(side(/*1+4*/2,/*2*/0,nx,ny)==1 && sn<8) flag=108;

else if(side(/*1+5*/6,/*0*/2,nx,ny)==1 && sn<8) flag=109;
else if(side(/*1+3*/3,1,nx,ny)==1 && sn<8) flag=110;
else if(side(/*1+5*/6,1,nx,ny)==1 && sn<8) flag=111;
else if(side(/*1+2*/4,1,nx,ny)==1 && sn<8) flag=112;
else if(side(/*1+5*/6,/*2*/0,nx,ny)==1 && sn<8) flag=113;
else if(side(/*1+1*/5,1,nx,ny)==1 && sn<8) flag=114;

/*999*/
#if N1>=2
/* outer */
else if(side_('A',0,nx,ny)==1) flag=115;
else if(side_('a',/*1*/0,nx,ny)==1) flag=116;
else if(side_('A',2,nx,ny)==1) flag=117;
else if(side_('a',/*3*/2,nx,ny)==1) flag=118;
else if(side_('A',3,nx,ny)==1) flag=119;
else if(side_('a',/*4*/3,nx,ny)==1) flag=120;
else if(side_('A',1,nx,ny)==1) flag=121;
else if(side_('a',/*2*/1,nx,ny)==1) flag=122;

else if(side_('F',0,nx,ny)==1) flag=123;
else if(side_('f',/*1*/0,nx,ny)==1) flag=124;
else if(side_('F',2,nx,ny)==1) flag=125;
else if(side_('f',/*3*/2,nx,ny)==1) flag=126;
else if(side_('F',3,nx,ny)==1) flag=127;
else if(side_('f',/*4*/3,nx,ny)==1) flag=128;
else if(side_('F',1,nx,ny)==1) flag=129;
else if(side_('f',/*2*/1,nx,ny)==1) flag=130;

else if(side_('B',0,nx,ny)==1) flag=131;
else if(side_('b',/*1*/0,nx,ny)==1) flag=132;
else if(side_('B',2,nx,ny)==1) flag=133;
else if(side_('b',/*3*/2,nx,ny)==1) flag=134;
else if(side_('B',3,nx,ny)==1) flag=135;
else if(side_('b',/*4*/3,nx,ny)==1) flag=136;
else if(side_('B',1,nx,ny)==1) flag=137;
else if(side_('b',/*2*/1,nx,ny)==1) flag=138;

else if(side_('D',0,nx,ny)==1) flag=139;
else if(side_('d',/*1*/0,nx,ny)==1) flag=140;
else if(side_('D',2,nx,ny)==1) flag=141;
else if(side_('d',/*3*/2,nx,ny)==1) flag=142;

```

```
else if(side_('D',3,nx,ny)==1) flag=143;
else if(side_('d',/*4*/3,nx,ny)==1) flag=144;
else if(side_('D',1,nx,ny)==1) flag=145;
else if(side_('d',/*2*/1,nx,ny)==1) flag=146;
```

```
else if(side_('C',0,nx,ny)==1) flag=147;
else if(side_('c',/*1*/0,nx,ny)==1) flag=148;
else if(side_('C',2,nx,ny)==1) flag=149;
else if(side_('c',/*3*/2,nx,ny)==1) flag=150;
else if(side_('C',3,nx,ny)==1) flag=151;
else if(side_('c',/*4*/3,nx,ny)==1) flag=152;
else if(side_('C',1,nx,ny)==1) flag=153;
else if(side_('c',/*2*/1,nx,ny)==1) flag=154;
```

```
else if(side_('E',0,nx,ny)==1) flag=155;
else if(side_('e',/*1*/0,nx,ny)==1) flag=156;
else if(side_('E',2,nx,ny)==1) flag=157;
else if(side_('e',/*3*/2,nx,ny)==1) flag=158;
else if(side_('E',3,nx,ny)==1) flag=159;
else if(side_('e',/*4*/3,nx,ny)==1) flag=160;
else if(side_('E',1,nx,ny)==1) flag=161;
else if(side_('e',/*2*/1,nx,ny)==1) flag=162;
```

```
/* inner */
```

```
/* 9, 10 */
```

```
else if(side_('a',9,nx,ny)==1) flag=1001;
else if(side_('b',9,nx,ny)==1) flag=1001;
else if(side_('c',9,nx,ny)==1) flag=1001;
else if(side_('d',9,nx,ny)==1) flag=1001;
else if(side_('e',9,nx,ny)==1) flag=1001;
else if(side_('f',9,nx,ny)==1) flag=1001;
else if(side_('a',10,nx,ny)==1) flag=1002;
else if(side_('b',10,nx,ny)==1) flag=1002;
else if(side_('c',10,nx,ny)==1) flag=1002;
else if(side_('d',10,nx,ny)==1) flag=1002;
else if(side_('e',10,nx,ny)==1) flag=1002;
else if(side_('f',10,nx,ny)==1) flag=1002;
```

```
/* A */
```

```
else if(side_('a',/*9*/5,nx,ny)==1) flag=1003;
else if(side_('e',/*6*/8,nx,ny)==1) flag=1004;
else if(side_('a',7,nx,ny)==1) flag=1005;
else if(side_('c',/*6*/8,nx,ny)==1) flag=1006;
else if(side_('a',/*8*/6,nx,ny)==1) flag=1007;
else if(side_('b',/*6*/8,nx,ny)==1) flag=1008;
else if(side_('a',/*6*/8,nx,ny)==1) flag=1009;
```

```
else if(side_('d',/*6*/8,nx,ny)==1) flag=1010;
```

```
/* F */
```

```
else if(side_('b',/*8*/6,nx,ny)==1) flag=1011;
```

```
else if(side_('f',/*6*/8,nx,ny)==1) flag=1012;
```

```
else if(side_('d',/*8*/6,nx,ny)==1) flag=1013;
```

```
else if(side_('f',/*8*/6,nx,ny)==1) flag=1014;
```

```
else if(side_('e',/*8*/6,nx,ny)==1) flag=1015;
```

```
else if(side_('f',/*9*/5,nx,ny)==1) flag=1016;
```

```
else if(side_('c',/*8*/6,nx,ny)==1) flag=1017;
```

```
else if(side_('f',7,nx,ny)==1) flag=1018;
```

```
/* B~E */
```

```
else if(side_('b',/*9*/5,nx,ny)==1) flag=1019;
```

```
else if(side_('e',7,nx,ny)==1) flag=1020;
```

```
else if(side_('b',7,nx,ny)==1) flag=1021;
```

```
else if(side_('c',/*9*/5,nx,ny)==1) flag=1022;
```

```
else if(side_('c',7,nx,ny)==1) flag=1023;
```

```
else if(side_('d',/*9*/5,nx,ny)==1) flag=1024;
```

```
else if(side_('d',7,nx,ny)==1) flag=1025;
```

```
else if(side_('e',/*9*/5,nx,ny)==1) flag=1026;
```

```
#endif
```

```
else flag=-1;
```

```
if(flag>0){
```

```
if(flag==101){
```

```
dlt=1*(n-1)-nx;nx=1*(n-1);ny=1*(n-1)+dlt;;
```

```
}
```

```
else if(flag==102){
```

```
dlt=2*(n-1)-ny;nx=0*(n-1)+dlt;ny=1*(n-1);;
```

```
}
```

```
else if(flag==103){
```

```
dlt=1*(n-1)-ny;nx=1*(n-1);ny=2*(n-1)+dlt;;
```

```
}
```

```
else if(flag==104){
```

```
dlt=3*(n-1)-ny;nx=0*(n-1);ny=0*(n-1)+dlt;;
```

```
}
```

```
else if(flag==105){
```

```
dlt=nx-0*(n-1);nx=1*(n-1);ny=3*(n-1)+dlt;;
```

```
}
```

```
else if(flag==106){
```

```
dlt=ny-3*(n-1);nx=0*(n-1)+dlt;ny=0*(n-1);;
```

```
}
```

```
else if(flag==107){
```

```

nx=nx;ny=0*(n-1);;
}
else if(flag==108){
nx=nx;ny=4*(n-1);;
}

else if(flag==109){
dlt=nx-2*(n-1);nx=2*(n-1);ny=1*(n-1)+dlt;;
}
else if(flag==110){
dlt=ny-1*(n-1);nx=2*(n-1)+dlt;ny=1*(n-1);;
}
else if(flag==111){
dlt=1*(n-1)-ny;nx=2*(n-1);ny=2*(n-1)+dlt;;
}
else if(flag==112){
dlt=3*(n-1)-ny;nx=3*(n-1);ny=0*(n-1)+dlt;;
}
else if(flag==113){
dlt=3*(n-1)-nx;nx=2*(n-1);ny=3*(n-1)+dlt;;
}
else if(flag==114){
dlt=4*(n-1)-ny;nx=2*(n-1)+dlt;ny=0*(n-1);;
}

else if((qx=0)>-1 && flag==115){ /* A */
dlt=nx-(*0*/qx*(n-1)+(N1-1));nx=4*(n-1);ny=dlt;
}
else if(flag==116){
dlt=ny;nx=*0*/qx*(n-1)+N1-1+dlt;ny=N1-1;
}
else if(flag==117){
dlt=*1*/(qx+1)*(n-1)-(N1-1)-nx;nx=4*(n-1);ny=2*(N2-1)+dlt;
}
else if(flag==118){
dlt=3*(N2-1)-ny;nx=*0*/qx*(n-1)+N1-1+dlt;ny=1*(n-1)-(N1-1);
}
else if(flag==119){
dlt=1*(n-1)-(N1-1)-ny;nx=4*(n-1);ny=3*(N2-1)+dlt;
}
else if(flag==120){
dlt=4*(N2-1)-ny;nx=*0*/qx*(n-1)+N1-1;ny=N1-1+dlt;
}
else if(flag==121){
dlt=ny-(N1-1);nx=4*(n-1);ny=1*(N2-1)+dlt;
}

```



```

else if(flag==122){
dlt=ny-1*(N2-1);nx=/*1*/(qx+1)*(n-1)-(N1-1);ny=N1-1+dlt;
}

else if((qx=2)>-1 && flag==123){ /* F */
dlt=nx-(*2*/qx*(n-1)+(N1-1));nx=9*(n-1);ny=dlt;
}
else if(flag==124){
dlt=ny;nx=/*2*/qx*(n-1)+N1-1+dlt;ny=N1-1;
}
else if(flag==125){
dlt=/*3*/(qx+1)*(n-1)-(N1-1)-nx;nx=9*(n-1);ny=2*(N2-1)+dlt;
}
else if(flag==126){
dlt=3*(N2-1)-ny;nx=/*2*/qx*(n-1)+N1-1+dlt;ny=1*(n-1)-(N1-1);
}
else if(flag==127){
dlt=1*(n-1)-(N1-1)-ny;nx=9*(n-1);ny=3*(N2-1)+dlt;
}
else if(flag==128){
dlt=4*(N2-1)-ny;nx=/*2*/qx*(n-1)+N1-1;ny=0*(n-1)+(N1-1)+dlt;
}
else if(flag==129){
dlt=ny-(N1-1);nx=9*(n-1);ny=1*(N2-1)+dlt;
}
else if(flag==130){
dlt=ny-1*(N2-1);nx=/*3*/(qx+1)*(n-1)-(N1-1);ny=(N1-1)+dlt;
}

else if((qx=0)>-1 && flag==131){ /* B */
dlt=nx-(1*(n-1)+(N1-1));nx=5*(n-1);ny=dlt;
}
else if(flag==132){
dlt=ny;nx=1*(n-1)+N1-1+dlt;ny=qx*(n-1)+N1-1;
}
else if(flag==133){
dlt=2*(n-1)-(N1-1)-nx;nx=5*(n-1);ny=2*(N2-1)+dlt;
}
else if(flag==134){
dlt=3*(N2-1)-ny;nx=1*(n-1)+N1-1+dlt;ny=(qx+1)*(n-1)-(N1-1);
}
else if(flag==135){
dlt=(qx+1)*(n-1)-(N1-1)-ny;nx=5*(n-1);ny=3*(N2-1)+dlt;
}
else if(flag==136){
dlt=4*(N2-1)-ny;nx=n-1+N1-1;ny=qx*(n-1)+(N1-1)+dlt;
}

```

```

}
else if(flag==137){
dlt=ny-(qx*(n-1)+(N1-1));nx=5*(n-1);ny=N2-1+dlt;
}
else if(flag==138){
dlt=ny-(N2-1);nx=2*(n-1)-(N1-1);ny=qx*(n-1)+(N1-1)+dlt;
}

else if((qx=2)>-1 && flag==139){ /* D */
dlt=nx-(1*(n-1)+(N1-1));nx=7*(n-1);ny=dlt;
}
else if(flag==140){
dlt=ny;nx=1*(n-1)+N1-1+dlt;ny=qx*(n-1)+N1-1;
}
else if(flag==141){
dlt=2*(n-1)-(N1-1)-nx;nx=7*(n-1);ny=2*(N2-1)+dlt;
}
else if(flag==142){
dlt=3*(N2-1)-ny;nx=1*(n-1)+N1-1+dlt;ny=(qx+1)*(n-1)-(N1-1);
}
else if(flag==143){
dlt=(qx+1)*(n-1)-(N1-1)-ny;nx=7*(n-1);ny=3*(N2-1)+dlt;
}
else if(flag==144){
dlt=4*(N2-1)-ny;nx=n-1+N1-1;ny=qx*(n-1)+(N1-1)+dlt;
}
else if(flag==145){
dlt=ny-(qx*(n-1)+(N1-1));nx=7*(n-1);ny=N2-1+dlt;
}
else if(flag==146){
dlt=ny-(N2-1);nx=2*(n-1)-(N1-1);ny=qx*(n-1)+(N1-1)+dlt;
}

else if((qx=1)>-1 && flag==147){ /* C */
dlt=nx-(1*(n-1)+(N1-1));nx=6*(n-1);ny=dlt;
}
else if(flag==148){
dlt=ny;nx=1*(n-1)+N1-1+dlt;ny=qx*(n-1)+N1-1;
}
else if(flag==149){
dlt=2*(n-1)-(N1-1)-nx;nx=6*(n-1);ny=2*(N2-1)+dlt;
}
else if(flag==150){
dlt=3*(N2-1)-ny;nx=1*(n-1)+N1-1+dlt;ny=(qx+1)*(n-1)-(N1-1);
}
else if(flag==151){

```

```

dlt=(qx+1)*(n-1)-(N1-1)-ny;nx=6*(n-1);ny=3*(N2-1)+dlt;
}
else if(flag==152){
dlt=4*(N2-1)-ny;nx=n-1+N1-1;ny=qx*(n-1)+(N1-1)+dlt;
}
else if(flag==153){
dlt=ny-(qx*(n-1)+(N1-1));nx=6*(n-1);ny=N2-1+dlt;
}
else if(flag==154){
dlt=ny-(N2-1);nx=2*(n-1)-(N1-1);ny=qx*(n-1)+(N1-1)+dlt;
}

else if((qx=3)>-1 && flag==155){ /* E */
dlt=nx-(1*(n-1)+(N1-1));nx=8*(n-1);ny=dlt;
}
else if(flag==156){
dlt=ny;nx=1*(n-1)+N1-1+dlt;ny=qx*(n-1)+N1-1;
}
else if(flag==157){
dlt=2*(n-1)-(N1-1)-nx;nx=8*(n-1);ny=2*(N2-1)+dlt;
}
else if(flag==158){
dlt=3*(N2-1)-ny;nx=1*(n-1)+N1-1+dlt;ny=(qx+1)*(n-1)-(N1-1);
}
else if(flag==159){
dlt=(qx+1)*(n-1)-(N1-1)-ny;nx=8*(n-1);ny=3*(N2-1)+dlt;
}
else if(flag==160){
dlt=4*(N2-1)-ny;nx=n-1+N1-1;ny=qx*(n-1)+(N1-1)+dlt;
}
else if(flag==161){
dlt=ny-(qx*(n-1)+(N1-1));nx=8*(n-1);ny=N2-1+dlt;
}
else if(flag==162){
dlt=ny-(N2-1);nx=2*(n-1)-(N1-1);ny=qx*(n-1)+(N1-1)+dlt;
}

else if(flag==1001){
nx=nx;ny=4*(N2-1);
}
else if(flag==1002){
nx=nx;ny=0*(N2-1);
}

else if(flag==1003){
dlt=1*(N2-1)-ny;nx=8*(n-1)-(N1-1);ny=3*(N2-1)+dlt;

```

```

}
else if(flag==1004){
dlt=4*(N2-1)-ny;nx=4*(n-1)-(N1-1);ny=0*(N2-1)+dlt;
}
else if(flag==1005){
dlt=3*(N2-1)-ny;nx=6*(n-1)-(N1-1);ny=3*(N2-1)+dlt;
}
else if(flag==1006){
dlt=4*(N2-1)-ny;nx=4*(n-1)-(N1-1);ny=2*(N2-1)+dlt;
}
else if(flag==1007){
dlt=2*(N2-1)-ny;nx=5*(n-1)-(N1-1);ny=3*(N2-1)+dlt;
}
else if(flag==1008){
dlt=4*(N2-1)-ny;nx=4*(n-1)-(N1-1);ny=1*(N2-1)+dlt;
}
else if(flag==1009){
dlt=4*(N2-1)-ny;nx=7*(n-1)-(N1-1);ny=3*(N2-1)+dlt;
}
else if(flag==1010){
dlt=4*(N2-1)-ny;nx=4*(n-1)-(N1-1);ny=3*(N2-1)+dlt;
}

else if(flag==1011){
dlt=2*(N2-1)-ny;nx=9*(n-1)-(N1-1);ny=3*(N2-1)+dlt;
}
else if(flag==1012){
dlt=4*(N2-1)-ny;nx=5*(n-1)-(N1-1);ny=1*(N2-1)+dlt;
}
else if(flag==1013){
dlt=2*(N2-1)-ny;nx=9*(n-1)-(N1-1);ny=1*(N2-1)+dlt;
}
else if(flag==1014){
dlt=2*(N2-1)-ny;nx=7*(n-1)-(N1-1);ny=1*(N2-1)+dlt;
}
else if(flag==1015){
dlt=2*(N2-1)-ny;nx=9*(n-1)-(N1-1);ny=0*(N2-1)+dlt;
}
else if(flag==1016){
dlt=1*(N2-1)-ny;nx=8*(n-1)-(N1-1);ny=1*(N2-1)+dlt;
}
else if(flag==1017){
dlt=2*(N2-1)-ny;nx=9*(n-1)-(N1-1);ny=2*(N2-1)+dlt;
}
else if(flag==1018){
dlt=3*(N2-1)-ny;nx=6*(n-1)-(N1-1);ny=1*(N2-1)+dlt;
}

```

```

}

else if(flag==1019){
dlt=1*(N2-1)-ny;nx=8*(n-1)-(N1-1);ny=2*(N2-1)+dlt;
}
else if(flag==1020){
dlt=3*(N2-1)-ny;nx=5*(n-1)-(N1-1);ny=0*(N2-1)+dlt;
}
else if(flag==1021){
dlt=3*(N2-1)-ny;nx=6*(n-1)-(N1-1);ny=0*(N2-1)+dlt;
}
else if(flag==1022){
dlt=1*(N2-1)-ny;nx=5*(n-1)-(N1-1);ny=2*(N2-1)+dlt;
}
else if(flag==1023){
dlt=3*(N2-1)-ny;nx=7*(n-1)-(N1-1);ny=0*(N2-1)+dlt;
}
else if(flag==1024){
dlt=1*(N2-1)-ny;nx=6*(n-1)-(N1-1);ny=2*(N2-1)+dlt;
}
else if(flag==1025){
dlt=3*(N2-1)-ny;nx=8*(n-1)-(N1-1);ny=0*(N2-1)+dlt;
}
else if(flag==1026){
dlt=1*(N2-1)-ny;nx=7*(n-1)-(N1-1);ny=2*(N2-1)+dlt;
}

nx+=sn*(n-1);
putpixel(nx,ny,pcolor);
}/**if(flag>0)**/
}/**else(0)**/
}/** putpixel_ **/

int getpixel_(int x,int y,int nx,int ny)
{
int i,n,qx,flag,val,dsn,dlt;

n=RESO;

x-=sn*(n-1);
nx-=sn*(n-1);

/*if(nx<0 || ny<0) return 1;*/ /* here:0(old) or 1 */

if(v(*1+0*/1,*0*/3,x,y)) flag=1; /* !nx, !ny */

```

```

else if(v(/*1+3*/4,0,x,y)) flag=2;
else if(v(/*1+0*/1,/*2*/1,x,y)) flag=3;
else if(v(/*1+1*/5,/*0*/3,x,y)) flag=4;
else if(v(/*1+0*/1,/*3*/0,x,y)) flag=5;
else if(v(/*1+2*/5,0,x,y)) flag=6;
else if(v(/*1+5*/6,/*1*/2,x,y)) flag=7;
else if(v(/*1+3*/4,1,x,y)) flag=8;
else if(v(/*1+5*/6,/*2*/1,x,y)) flag=9;
else if(v(/*1+2*/5,1,x,y)) flag=10;
else if(v(/*1+5*/6,/*3*/0,x,y)) flag=11;
else if(v(/*1+1*/5,/*1*/2,x,y)) flag=12;

else if(side(/*1+0*/1,/*0*/2,x,y)==1 && sn<8) flag=101;
else if(side(/*1+3*/3,3,x,y)==1 && sn<8) flag=102;
else if(side(/*1+0*/1,3,x,y)==1 && sn<8) flag=103;
else if(side(/*1+2*/4,3,x,y)==1 && sn<8) flag=104;
else if(side(/*1+0*/1,/*2*/0,x,y)==1 && sn<8) flag=105;
else if(side(/*1+1*/5,3,x,y)==1 && sn<8) flag=106;

else if(side(/*1+1*/5,/*0*/2,x,y)==1 && sn<8) flag=107;
else if(side(/*1+4*/2,/*2*/0,x,y)==1 && sn<8) flag=108;

else if(side(/*1+5*/6,/*0*/2,x,y)==1 && sn<8) flag=109;
else if(side(/*1+3*/3,1,x,y)==1 && sn<8) flag=110;
else if(side(/*1+5*/6,1,x,y)==1 && sn<8) flag=111;
else if(side(/*1+2*/4,1,x,y)==1 && sn<8) flag=112;
else if(side(/*1+5*/6,/*2*/0,x,y)==1 && sn<8) flag=113;
else if(side(/*1+1*/5,1,x,y)==1 && sn<8) flag=114;

/*999*/
#if N1>=2
else if(v_('a',0,x,y)) flag=13;
else if(v_('b',0,x,y)) flag=13;
else if(v_('c',0,x,y)) flag=13;
else if(v_('d',0,x,y)) flag=13;
else if(v_('e',0,x,y)) flag=13;
else if(v_('f',0,x,y)) flag=13;
else if(v_('a',4,x,y)) flag=14;
else if(v_('b',4,x,y)) flag=14;
else if(v_('c',4,x,y)) flag=14;
else if(v_('d',4,x,y)) flag=14;
else if(v_('e',4,x,y)) flag=14;
else if(v_('f',4,x,y)) flag=14;
else if(v_('a',0+5,x,y)) flag=13;
else if(v_('b',0+5,x,y)) flag=13;
else if(v_('c',0+5,x,y)) flag=13;

```

```

else if(v_('d',0+5,x,y)) flag=13;
else if(v_('e',0+5,x,y)) flag=13;
else if(v_('f',0+5,x,y)) flag=13;
else if(v_('a',4+5,x,y)) flag=14;
else if(v_('b',4+5,x,y)) flag=14;
else if(v_('c',4+5,x,y)) flag=14;
else if(v_('d',4+5,x,y)) flag=14;
else if(v_('e',4+5,x,y)) flag=14;
else if(v_('f',4+5,x,y)) flag=14;

/* outer */
else if(side_('A',0,x,y)==1) flag=115;
else if(side_('a',/*1*/0,x,y)==1) flag=116;
else if(side_('A',2,x,y)==1) flag=117;
else if(side_('a',/*3*/2,x,y)==1) flag=118;
else if(side_('A',3,x,y)==1) flag=119;
else if(side_('a',/*4*/3,x,y)==1) flag=120;
else if(side_('A',1,x,y)==1) flag=121;
else if(side_('a',/*2*/1,x,y)==1) flag=122;

else if(side_('F',0,x,y)==1) flag=123;
else if(side_('f',/*1*/0,x,y)==1) flag=124;
else if(side_('F',2,x,y)==1) flag=125;
else if(side_('f',/*3*/2,x,y)==1) flag=126;
else if(side_('F',3,x,y)==1) flag=127;
else if(side_('f',/*4*/3,x,y)==1) flag=128;
else if(side_('F',1,x,y)==1) flag=129;
else if(side_('f',/*2*/1,x,y)==1) flag=130;

else if(side_('B',0,x,y)==1) flag=131;
else if(side_('b',/*1*/0,x,y)==1) flag=132;
else if(side_('B',2,x,y)==1) flag=133;
else if(side_('b',/*3*/2,x,y)==1) flag=134;
else if(side_('B',3,x,y)==1) flag=135;
else if(side_('b',/*4*/3,x,y)==1) flag=136;
else if(side_('B',1,x,y)==1) flag=137;
else if(side_('b',/*2*/1,x,y)==1) flag=138;

else if(side_('D',0,x,y)==1) flag=139;
else if(side_('d',/*1*/0,x,y)==1) flag=140;
else if(side_('D',2,x,y)==1) flag=141;
else if(side_('d',/*3*/2,x,y)==1) flag=142;
else if(side_('D',3,x,y)==1) flag=143;
else if(side_('d',/*4*/3,x,y)==1) flag=144;
else if(side_('D',1,x,y)==1) flag=145;
else if(side_('d',/*2*/1,x,y)==1) flag=146;

```

```
else if(side_('C',0,x,y)==1) flag=147;
else if(side_('c',/*1*/0,x,y)==1) flag=148;
else if(side_('C',2,x,y)==1) flag=149;
else if(side_('c',/*3*/2,x,y)==1) flag=150;
else if(side_('C',3,x,y)==1) flag=151;
else if(side_('c',/*4*/3,x,y)==1) flag=152;
else if(side_('C',1,x,y)==1) flag=153;
else if(side_('c',/*2*/1,x,y)==1) flag=154;
```

```
else if(side_('E',0,x,y)==1) flag=155;
else if(side_('e',/*1*/0,x,y)==1) flag=156;
else if(side_('E',2,x,y)==1) flag=157;
else if(side_('e',/*3*/2,x,y)==1) flag=158;
else if(side_('E',3,x,y)==1) flag=159;
else if(side_('e',/*4*/3,x,y)==1) flag=160;
else if(side_('E',1,x,y)==1) flag=161;
else if(side_('e',/*2*/1,x,y)==1) flag=162;
```

```
/* inner */
```

```
/* 9, 10 */
```

```
else if(side_('a',9,x,y)==1) flag=1001;
else if(side_('b',9,x,y)==1) flag=1001;
else if(side_('c',9,x,y)==1) flag=1001;
else if(side_('d',9,x,y)==1) flag=1001;
else if(side_('e',9,x,y)==1) flag=1001;
else if(side_('f',9,x,y)==1) flag=1001;
else if(side_('a',10,x,y)==1) flag=1002;
else if(side_('b',10,x,y)==1) flag=1002;
else if(side_('c',10,x,y)==1) flag=1002;
else if(side_('d',10,x,y)==1) flag=1002;
else if(side_('e',10,x,y)==1) flag=1002;
else if(side_('f',10,x,y)==1) flag=1002;
```

```
/* A */
```

```
else if(side_('a',/*9*/5,x,y)==1) flag=1003;
else if(side_('e',/*6*/8,x,y)==1) flag=1004;
else if(side_('a',7,x,y)==1) flag=1005;
else if(side_('c',/*6*/8,x,y)==1) flag=1006;
else if(side_('a',/*8*/6,x,y)==1) flag=1007;
else if(side_('b',/*6*/8,x,y)==1) flag=1008;
else if(side_('a',/*6*/8,x,y)==1) flag=1009;
else if(side_('d',/*6*/8,x,y)==1) flag=1010;
```

```
/* F */
```

```
else if(side_('b',/*8*/6,x,y)==1) flag=1011;
```



```

else if(side_('f',/*6*/8,x,y)==1) flag=1012;
else if(side_('d',/*8*/6,x,y)==1) flag=1013;
else if(side_('f',/*8*/6,x,y)==1) flag=1014;
else if(side_('e',/*8*/6,x,y)==1) flag=1015;
else if(side_('f',/*9*/5,x,y)==1) flag=1016;
else if(side_('c',/*8*/6,x,y)==1) flag=1017;
else if(side_('f',7,x,y)==1) flag=1018;

/* B~E */
else if(side_('b',/*9*/5,x,y)==1) flag=1019;
else if(side_('e',7,x,y)==1) flag=1020;
else if(side_('b',7,x,y)==1) flag=1021;
else if(side_('c',/*9*/5,x,y)==1) flag=1022;
else if(side_('c',7,x,y)==1) flag=1023;
else if(side_('d',/*9*/5,x,y)==1) flag=1024;
else if(side_('d',7,x,y)==1) flag=1025;
else if(side_('e',/*9*/5,x,y)==1) flag=1026;
#endif

else flag=0;

X=nx;Y=ny;
jmpflag=0;
dsn=0;
/*goto end;*/ /* here -> 0 or 1 in arrayreset() */

if(flag==1){ /* v(1,3, */
    if(nh(x,y,nx,ny,1)) {x=1*(n-1);y=2*(n-1);X=x+1;Y=y;jmpflag=1;;}
else if(nx<0*(n-1) || ny>1*(n-1)) return 0;;
}
else if(flag==2){ /* v(4,0, */
if(nx<1*(n-1)) return 0;;
}
else if(flag==3){ /* v(1,1, */
if(ny<0*(n-1)) return 0;;
}
else if(flag==4){ /* v(5,3, */
    if(nh(x,y,nx,ny,1)) {x=1*(n-1);y=0*(n-1);X=x;Y=y/*-*/+1;jmpflag=1;;}
else if(nx<1*(n-1) || ny>4*(n-1)) return 0;;
}
else if(flag==5){ /* v(1,0, */
    if(nh(x,y,nx,ny,3)) {x=1*(n-1);y=3*(n-1);X=x+1;Y=y;jmpflag=1;;}
else if(nx<0*(n-1) || ny<0*(n-1)) return 0;;
}
else if(flag==6){ /* v(5,0, */
if(nx<1*(n-1)) return 0;;

```

```

}
else if(flag==7){ /* v(6,2, */
    if(nh(x,y,nx,ny,0)) {x=2*(n-1);y=2*(n-1);X=x-1;Y=y;jmpflag=1;;}
else if(nx>3*(n-1) || ny>1*(n-1)) return 0;;
}
else if(flag==8){ /* v(4,1, */
if(nx>2*(n-1)) return 0;;
}
else if(flag==9){ /* v(6,1, */
    if(nh(x,y,nx,ny,0)) {x=2*(n-1);y=3*(n-1);X=x-1;Y=y;jmpflag=1;;}
else if(nx>3*(n-1) || ny<0*(n-1)) return 0;;
}
else if(flag==10){ /* v(5,1, */
if(nx>2*(n-1)) return 0;;
}
else if(flag==11){ /* v(6,0, */
if(ny<0*(n-1)) return 0;;
}
else if(flag==12){ /* v(5,2, */
    if(nh(x,y,nx,ny,0)) {x=2*(n-1);y=0*(n-1);X=x;Y=y/*-*/+1;jmpflag=1;;}
else if(nx>2*(n-1) || ny>4*(n-1)) return 0;;
}

else if(flag==13){ /* v_('a',0, */
    if(nh(x,y,nx,ny,3)) {x=x;y=4*(N2-1);X=x;Y=y-1;jmpflag=1;}
}
else if(flag==14){ /* v_('a',4, */
    if(nh(x,y,nx,ny,1)) {x=x;y=0*(N2-1);X=x;Y=y+1;jmpflag=1;}
}

else if(flag==101){ /* side(1,2, */
    if(nh(x,y,nx,ny,1)==1){
dlt=1*(n-1)-x;x=1*(n-1);y=1*(n-1)+dlt;;
rot(1,-1);jmpflag=101;;
}
}
else if(flag==102){ /* side(3,3, */
    if(nh(x,y,nx,ny,2)==1){
dlt=2*(n-1)-y;x=0*(n-1)+dlt;y=1*(n-1);;
rot(2,1);jmpflag=102;;
}
}
else if(flag==103){ /* side(1,3, */
    if(nh(x,y,nx,ny,2)==1){
dlt=1*(n-1)-y;x=1*(n-1);y=2*(n-1)+dlt;;

```

```

rot(2,-2);jmpflag=103;;
}
}
else if(flag==104){ /* side(4,3, */
    if(nh(x,y,nx,ny,2)==1){
dlt=3*(n-1)-y;x=0*(n-1);y=0*(n-1)+dlt;;
rot(2,-2);jmpflag=104;;
}
}
else if(flag==105){ /* side(1,0, */
    if(nh(x,y,nx,ny,3)==1){
dlt=x-0*(n-1);x=1*(n-1);y=3*(n-1)+dlt;;
rot(3,1);jmpflag=105;;
}
}
else if(flag==106){ /* side(5,3, */
    if(nh(x,y,nx,ny,2)==1){
dlt=y-3*(n-1);x=0*(n-1)+dlt;y=0*(n-1);;
rot(2,-1);jmpflag=106;;
}
}

else if(flag==107){ /* side(5,2, */
    if(nh(x,y,nx,ny,1)==1){
x=x;y=0*(n-1);;
rot(1,0);jmpflag=107;;
}
}
else if(flag==108){ /* side(2,0, */
    if(nh(x,y,nx,ny,3)==1){
x=x;y=4*(n-1);;
rot(3,0);jmpflag=108;;
}
}

else if(flag==109){ /* side(6,2, */
    if(nh(x,y,nx,ny,1)==1){
dlt=x-2*(n-1);x=2*(n-1);y=1*(n-1)+dlt;;
rot(1,1);jmpflag=109;;
}
}
else if(flag==110){ /* side(3,1, */
    if(nh(x,y,nx,ny,0)==1){
dlt=y-1*(n-1);x=2*(n-1)+dlt;y=1*(n-1);;
rot(0,-1);jmpflag=110;;
}
}

```

```

}
else if(flag==111){ /* side(6,1, */
    if(nh(x,y,nx,ny,0)==1){
dlt=1*(n-1)-y;x=2*(n-1);y=2*(n-1)+dlt;;
rot(0,2);jmpflag=111;;
}
}
else if(flag==112){ /* side(4,1, */
    if(nh(x,y,nx,ny,0)==1){
dlt=3*(n-1)-y;x=3*(n-1);y=0*(n-1)+dlt;;
rot(0,-2);jmpflag=112;;
}
}
else if(flag==113){ /* side(6,0, */
    if(nh(x,y,nx,ny,3)==1){
dlt=3*(n-1)-x;x=2*(n-1);y=3*(n-1)+dlt;;
rot(3,-1);jmpflag=113;;
}
}
else if(flag==114){ /* side(5,1, */
    if(nh(x,y,nx,ny,0)==1){
dlt=4*(n-1)-y;x=2*(n-1)+dlt;y=0*(n-1);;
rot(0,1);jmpflag=114;;
}
}

else if((qx=0)>-1 && flag==115){ /* side_('A',0, */
    if(nh(x,y,nx,ny,1)==1){
dlt=x-(*0*/qx*(n-1)+(N1-1));x=4*(n-1);y=0*(N2-1)+dlt;
rot(1,1);jmpflag=115;
}
}
else if(flag==116){ /* side_('a',0, */
    if(nh(x,y,nx,ny,0)==1){
dlt=y;x=*0*/qx*(n-1)+N1-1+dlt;y=0*(n-1)+N1-1;
rot(0,-1);jmpflag=116;
}
}
else if(flag==117){ /* side_('A',2, */
    if(nh(x,y,nx,ny,3)==1){
dlt=*1*/(qx+1)*(n-1)-(N1-1)-x;x=4*(n-1);y=2*(N2-1)+dlt;
rot(3,-1);jmpflag=117;
}
}
else if(flag==118){ /* side_('a',2, */
    if(nh(x,y,nx,ny,0)==1){

```

```

dlt=/*4*/3*(/*n*/N2-1)-y;x=/*0*/qx*(n-1)+N1-1+dlt;y=1*(n-1)-(N1-1);
rot(0,1);jmpflag=118;
}
}
else if(flag==119){ /* side_('A',3, */
    if(nh(x,y,nx,ny,0)==1){
dlt=1*(n-1)-(N1-1)-y;x=4*(n-1);y=3*(N2-1)+dlt;
rot(0,2);jmpflag=119;
}
}
else if(flag==120){ /* side_('a',3, */
    if(nh(x,y,nx,ny,0)==1){
dlt=4*(N2-1)-y;x=/*0*/qx*(n-1)+N1-1;y=N1-1+dlt;
rot(0,2);jmpflag=120;
}
}
else if(flag==121){ /* side_('A',1, */
    if(nh(x,y,nx,ny,2)==1){
dlt=y-(N1-1);x=4*(n-1);y=1*(N2-1)+dlt;
rot(2,0);jmpflag=121;
}
}
else if(flag==122){ /* side_('a',1, */
    if(nh(x,y,nx,ny,0)==1){
dlt=y-1*(N2-1);x=/*1*/(qx+1)*(n-1)-(N1-1);y=N1-1+dlt;
rot(0,0);jmpflag=122;
}
}

else if((qx=2)>-1 && flag==123){ /* side_('F',0, */
    if(nh(x,y,nx,ny,1)==1){
dlt=x-(/*2*/qx*(n-1)+(N1-1));x=9*(n-1);y=dlt;
rot(1,1);jmpflag=123;
}
}
else if(flag==124){ /* side_('f',0, */
    if(nh(x,y,nx,ny,0)==1){
dlt=y;x=/*2*/qx*(n-1)+N1-1+dlt;y=N1-1;
rot(0,-1);jmpflag=124;
}
}
else if(flag==125){ /* side_('F',2, */
    if(nh(x,y,nx,ny,3)==1){
dlt=/*3*/(qx+1)*(n-1)-(N1-1)-x;x=9*(n-1);y=2*(N2-1)+dlt;
rot(3,-1);jmpflag=125;
}
}

```

```

}
else if(flag==126){ /* side_('f',2, */
    if(nh(x,y,nx,ny,0)==1){
dlt=3*(N2-1)-y;x=/*2*/qx*(n-1)+N1-1+dlt;y=1*(n-1)-(N1-1);
rot(0,1);jmpflag=126;
}
}
else if(flag==127){ /* side_('F',3, */
    if(nh(x,y,nx,ny,0)==1){
dlt=1*(n-1)-(N1-1)-y;x=9*(n-1);y=3*(N2-1)+dlt;
rot(0,2);jmpflag=127;
}
}
else if(flag==128){ /* side_('f',3, */
    if(nh(x,y,nx,ny,0)==1){
dlt=4*(N2-1)-y;x=/*2*/qx*(n-1)+N1-1;y=0*(n-1)+(N1-1)+dlt;
rot(0,2);jmpflag=128;
}
}
else if(flag==129){ /* side_('F',1, */
    if(nh(x,y,nx,ny,2)==1){
dlt=y-(N1-1);x=9*(n-1);y=1*(N2-1)+dlt;
rot(2,0);jmpflag=129;
}
}
else if(flag==130){ /* side_('f',1, */
    if(nh(x,y,nx,ny,0)==1){
dlt=y-1*(N2-1);x=/*3*/(qx+1)*(n-1)-(N1-1);y=(N1-1)+dlt;
rot(0,0);jmpflag=130;
}
}

else if((qx=0)>-1 && flag==131){ /* side_('B',0, */
    if(nh(x,y,nx,ny,1)==1){
dlt=x-(1*(n-1)+(N1-1));x=5*(n-1);y=dlt;
rot(1,1);jmpflag=131;
}
}
else if(flag==132){ /* side_('b',0, */
    if(nh(x,y,nx,ny,0)==1){
dlt=y;x=1*(n-1)+N1-1+dlt;y=qx*(n-1)+N1-1;
rot(0,-1);jmpflag=132;
}
}
else if(flag==133){ /* side_('B',2, */
    if(nh(x,y,nx,ny,3)==1){

```

```

dlt=2*(n-1)-(N1-1)-x;x=5*(n-1);y=2*(N2-1)+dlt;
rot(3,-1);jmpflag=133;
}
}
else if(flag==134){ /* side_('b',2, */
    if(nh(x,y,nx,ny,0)==1){
dlt=3*(N2-1)-y;x=1*(n-1)+N1-1+dlt;y=(qx+1)*(n-1)-(N1-1);
rot(0,1);jmpflag=134;
}
}
else if(flag==135){ /* side_('B',3, */
    if(nh(x,y,nx,ny,0)==1){
dlt=(qx+1)*(n-1)-(N1-1)-y;x=5*(n-1);y=3*(N2-1)+dlt;
rot(0,2);jmpflag=135;
}
}
else if(flag==136){ /* side_('b',3, */
    if(nh(x,y,nx,ny,0)==1){
dlt=4*(N2-1)-y;x=n-1+N1-1;y=qx*(n-1)+(N1-1)+dlt;
rot(0,2);jmpflag=136;
}
}
else if(flag==137){ /* side_('B',1, */
    if(nh(x,y,nx,ny,2)==1){
dlt=y-(qx*(n-1)+(N1-1));x=5*(n-1);y=N2-1+dlt;
rot(2,0);jmpflag=137;
}
}
else if(flag==138){ /* side_('b',1, */
    if(nh(x,y,nx,ny,0)==1){
dlt=y-(N2-1);x=2*(n-1)-(N1-1);y=qx*(n-1)+(N1-1)+dlt;
rot(0,0);jmpflag=138;
}
}

else if((qx=2)>-1 && flag==139){ /* side_('D',0, */
    if(nh(x,y,nx,ny,1)==1){
dlt=x-(1*(n-1)+(N1-1));x=7*(n-1);y=dlt;
rot(1,1);jmpflag=139;
}
}
else if(flag==140){ /* side_('d',0, */
    if(nh(x,y,nx,ny,0)==1){
dlt=y;x=1*(n-1)+N1-1+dlt;y=qx*(n-1)+N1-1;
rot(0,-1);jmpflag=140;
}
}

```

```

}
else if(flag==141){ /* side_('D',2, */
    if(nh(x,y,nx,ny,3)==1){
dlt=2*(n-1)-(N1-1)-x;x=7*(n-1);y=2*(N2-1)+dlt;
rot(3,-1);jmpflag=141;
}
}
else if(flag==142){ /* side_('d',2, */
    if(nh(x,y,nx,ny,0)==1){
dlt=3*(N2-1)-y;x=1*(n-1)+N1-1+dlt;y=(qx+1)*(n-1)-(N1-1);
rot(0,1);jmpflag=142;
}
}
else if(flag==143){ /* side_('D',3, */
    if(nh(x,y,nx,ny,0)==1){
dlt=(qx+1)*(n-1)-(N1-1)-y;x=7*(n-1);y=3*(N2-1)+dlt;
rot(0,2);jmpflag=143;
}
}
else if(flag==144){ /* side_('d',3, */
    if(nh(x,y,nx,ny,0)==1){
dlt=4*(N2-1)-y;x=n-1+N1-1;y=qx*(n-1)+(N1-1)+dlt;
rot(0,2);jmpflag=144;
}
}
else if(flag==145){ /* side_('D',1, */
    if(nh(x,y,nx,ny,2)==1){
dlt=y-(qx*(n-1)+(N1-1));x=7*(n-1);y=N2-1+dlt;
rot(2,0);jmpflag=145;
}
}
else if(flag==146){ /* side_('d',1, */
    if(nh(x,y,nx,ny,0)==1){
dlt=y-(N2-1);x=2*(n-1)-(N1-1);y=qx*(n-1)+(N1-1)+dlt;
rot(0,0);jmpflag=146;
}
}

else if((qx=1)>-1 && flag==147){ /* side_('C',0, */
    if(nh(x,y,nx,ny,1)==1){
dlt=x-(1*(n-1)+(N1-1));x=6*(n-1);y=dlt;
rot(1,1);jmpflag=147;
}
}
else if(flag==148){ /* side_('c',0, */
    if(nh(x,y,nx,ny,0)==1){

```



```

dlt=y;x=1*(n-1)+N1-1+dlt;y=qx*(n-1)+N1-1;
rot(0,-1);jmpflag=148;
}
}
else if(flag==149){ /* side_('C',2, */
    if(nh(x,y,nx,ny,3)==1){
dlt=2*(n-1)-(N1-1)-x;x=6*(n-1);y=2*(N2-1)+dlt;
rot(3,-1);jmpflag=149;
}
}
else if(flag==150){ /* side_('c',2, */
    if(nh(x,y,nx,ny,0)==1){
dlt=3*(N2-1)-y;x=1*(n-1)+N1-1+dlt;y=(qx+1)*(n-1)-(N1-1);
rot(0,1);jmpflag=150;
}
}
else if(flag==151){ /* side_('C',3, */
    if(nh(x,y,nx,ny,0)==1){
dlt=(qx+1)*(n-1)-(N1-1)-y;x=6*(n-1);y=3*(N2-1)+dlt;
rot(0,2);jmpflag=151;
}
}
else if(flag==152){ /* side_('c',3, */
    if(nh(x,y,nx,ny,0)==1){
dlt=4*(N2-1)-y;x=n-1+N1-1;y=qx*(n-1)+(N1-1)+dlt;
rot(0,2);jmpflag=152;
}
}
else if(flag==153){ /* side_('C',1, */
    if(nh(x,y,nx,ny,2)==1){
dlt=y-(qx*(n-1)+(N1-1));x=6*(n-1);y=N2-1+dlt;
rot(2,0);jmpflag=153;
}
}
else if(flag==154){ /* side_('c',1, */
    if(nh(x,y,nx,ny,0)==1){
dlt=y-(N2-1);x=2*(n-1)-(N1-1);y=qx*(n-1)+(N1-1)+dlt;
rot(0,0);jmpflag=154;
}
}

else if((qx=3)>-1 && flag==155){ /* side_('E',0, */
    if(nh(x,y,nx,ny,1)==1){
dlt=x-(1*(n-1)+(N1-1));x=8*(n-1);y=dlt;
rot(1,1);jmpflag=155;
}
}

```

```

}
else if(flag==156){ /* side_('e',0, */
    if(nh(x,y,nx,ny,0)==1){
dlt=y;x=1*(n-1)+N1-1+dlt;y=qx*(n-1)+N1-1;
rot(0,-1);jmpflag=156;
}
}
else if(flag==157){ /* side_('E',2, */
    if(nh(x,y,nx,ny,3)==1){
dlt=2*(n-1)-(N1-1)-x;x=8*(n-1);y=2*(N2-1)+dlt;
rot(3,-1);jmpflag=157;
}
}
else if(flag==158){ /* side_('e',2, */
    if(nh(x,y,nx,ny,0)==1){
dlt=3*(N2-1)-y;x=1*(n-1)+N1-1+dlt;y=(qx+1)*(n-1)-(N1-1);
rot(0,1);jmpflag=158;
}
}
else if(flag==159){ /* side_('E',3, */
    if(nh(x,y,nx,ny,0)==1){
dlt=(qx+1)*(n-1)-(N1-1)-y;x=8*(n-1);y=3*(N2-1)+dlt;
rot(0,2);jmpflag=159;
}
}
else if(flag==160){ /* side_('e',3, */
    if(nh(x,y,nx,ny,0)==1){
dlt=4*(N2-1)-y;x=n-1+N1-1;y=qx*(n-1)+(N1-1)+dlt;
rot(0,2);jmpflag=160;
}
}
else if(flag==161){ /* side_('E',1, */
    if(nh(x,y,nx,ny,2)==1){
dlt=y-(qx*(n-1)+(N1-1));x=8*(n-1);y=N2-1+dlt;
rot(2,0);jmpflag=161;
}
}
else if(flag==162){ /* side_('e',1, */
    if(nh(x,y,nx,ny,0)==1){
dlt=y-(N2-1);x=2*(n-1)-(N1-1);y=qx*(n-1)+(N1-1)+dlt;
rot(0,0);jmpflag=162;
}
}

else if(flag==1001){ /* side_('a',9, */
    if(nh(x,y,nx,ny,3)==1){

```

```

x=x;y=4*(N2-1);
rot(3,0);jmpflag=1001;
}
}
else if(flag==1002){ /* side_('a',10, */
    if(nh(x,y,nx,ny,1)==1){
x=x;y=0*(N2-1);
rot(1,0);jmpflag=1002;
}
}

else if(flag==1003){ /* side_('a',5, */
    if(nh(x,y,nx,ny,2)==1){
dlt=1*(N2-1)-y;x=8*(n-1)-(N1-1);y=3*(N2-1)+dlt;
rot(2,2);jmpflag=1003;
}
}
else if(flag==1004){ /* side_('e',8, */
    if(nh(x,y,nx,ny,2)==1){
dlt=4*(N2-1)-y;x=4*(n-1)-(N1-1);y=0*(N2-1)+dlt;
rot(2,2);jmpflag=1004;
}
}
else if(flag==1005){ /* side_('a',7, */
    if(nh(x,y,nx,ny,2)==1){
dlt=3*(N2-1)-y;x=6*(n-1)-(N1-1);y=3*(N2-1)+dlt;
rot(2,2);jmpflag=1005;
}
}
else if(flag==1006){ /* side_('c',8, */
    if(nh(x,y,nx,ny,2)==1){
dlt=4*(N2-1)-y;x=4*(n-1)-(N1-1);y=2*(N2-1)+dlt;
rot(2,2);jmpflag=1006;
}
}
else if(flag==1007){ /* side_('a',6, */
    if(nh(x,y,nx,ny,2)==1){
dlt=2*(N2-1)-y;x=5*(n-1)-(N1-1);y=3*(N2-1)+dlt;
rot(2,2);jmpflag=1007;
}
}
else if(flag==1008){ /* side_('b',8, */
    if(nh(x,y,nx,ny,2)==1){
dlt=4*(N2-1)-y;x=4*(n-1)-(N1-1);y=1*(N2-1)+dlt;
rot(2,2);jmpflag=1008;
}
}

```

```

}
else if(flag==1009){ /* side_('a',8, */
    if(nh(x,y,nx,ny,2)==1){
dlt=4*(N2-1)-y;x=7*(n-1)-(N1-1);y=3*(N2-1)+dlt;
rot(2,2);jmpflag=1009;
}
}
else if(flag==1010){ /* side_('d',8, */
    if(nh(x,y,nx,ny,2)==1){
dlt=4*(N2-1)-y;x=4*(n-1)-(N1-1);y=3*(N2-1)+dlt;
rot(2,2);jmpflag=1010;
}
}

else if(flag==1011){ /* side_('b',6, */
    if(nh(x,y,nx,ny,2)==1){
dlt=2*(N2-1)-y;x=9*(n-1)-(N1-1);y=3*(N2-1)+dlt;
rot(2,2);jmpflag=1011;
}
}
else if(flag==1012){ /* side_('f',8, */
    if(nh(x,y,nx,ny,2)==1){
dlt=4*(N2-1)-y;x=5*(n-1)-(N1-1);y=1*(N2-1)+dlt;
rot(2,2);jmpflag=1012;
}
}
else if(flag==1013){ /* side_('d',6, */
    if(nh(x,y,nx,ny,2)==1){
dlt=2*(N2-1)-y;x=9*(n-1)-(N1-1);y=1*(N2-1)+dlt;
rot(2,2);jmpflag=1013;
}
}
else if(flag==1014){ /* side_('f',6, */
    if(nh(x,y,nx,ny,2)==1){
dlt=2*(N2-1)-y;x=7*(n-1)-(N1-1);y=1*(N2-1)+dlt;
rot(2,2);jmpflag=1014;
}
}
else if(flag==1015){ /* side_('e',6, */
    if(nh(x,y,nx,ny,2)==1){
dlt=2*(N2-1)-y;x=9*(n-1)-(N1-1);y=0*(N2-1)+dlt;
rot(2,2);jmpflag=1015;
}
}
else if(flag==1016){ /* side_('f',5, */
    if(nh(x,y,nx,ny,2)==1){

```

```

dlt=1*(N2-1)-y;x=8*(n-1)-(N1-1);y=1*(N2-1)+dlt;
rot(2,2);jmpflag=1016;
}
}
else if(flag==1017){ /* side_('c',6, */
    if(nh(x,y,nx,ny,2)==1){
dlt=2*(N2-1)-y;x=9*(n-1)-(N1-1);y=2*(N2-1)+dlt;
rot(2,2);jmpflag=1017;
}
}
else if(flag==1018){ /* side_('f',7, */
    if(nh(x,y,nx,ny,2)==1){
dlt=3*(N2-1)-y;x=6*(n-1)-(N1-1);y=1*(N2-1)+dlt;
rot(2,2);jmpflag=1018;
}
}

else if(flag==1019){ /* side_('b',5, */
    if(nh(x,y,nx,ny,2)==1){
dlt=1*(N2-1)-y;x=8*(n-1)-(N1-1);y=2*(N2-1)+dlt;
rot(2,2);jmpflag=1019;
}
}
else if(flag==1020){ /* side_('e',7, */
    if(nh(x,y,nx,ny,2)==1){
dlt=3*(N2-1)-y;x=5*(n-1)-(N1-1);y=0*(N2-1)+dlt;
rot(2,2);jmpflag=1020;
}
}
else if(flag==1021){ /* side_('b',7, */
    if(nh(x,y,nx,ny,2)==1){
dlt=3*(N2-1)-y;x=6*(n-1)-(N1-1);y=0*(N2-1)+dlt;
rot(2,2);jmpflag=1021;
}
}
else if(flag==1022){ /* side_('c',5, */
    if(nh(x,y,nx,ny,2)==1){
dlt=1*(N2-1)-y;x=5*(n-1)-(N1-1);y=2*(N2-1)+dlt;
rot(2,2);jmpflag=1022;
}
}
else if(flag==1023){ /* side_('c',7, */
    if(nh(x,y,nx,ny,2)==1){
dlt=3*(N2-1)-y;x=7*(n-1)-(N1-1);y=0*(N2-1)+dlt;
rot(2,2);jmpflag=1023;
}
}

```

```

}
else if(flag==1024){ /* side_('d',5, */
    if(nh(x,y,nx,ny,2)==1){
dlt=1*(N2-1)-y;x=6*(n-1)-(N1-1);y=2*(N2-1)+dlt;
rot(2,2);jmpflag=1024;
}
}
else if(flag==1025){ /* side_('d',7, */
    if(nh(x,y,nx,ny,2)==1){
dlt=3*(N2-1)-y;x=8*(n-1)-(N1-1);y=0*(N2-1)+dlt;
rot(2,2);jmpflag=1025;
}
}
else if(flag==1026){ /* side_('e',5, */
    if(nh(x,y,nx,ny,2)==1){
dlt=1*(N2-1)-y;x=7*(n-1)-(N1-1);y=2*(N2-1)+dlt;
rot(2,2);jmpflag=1026;
}
}

if(flag>=101 && flag<=1026 && jmpflag>100){
X=x+tmp0;
Y=y+tmp1;
}

end:
X_=x;
Y_=y;

X+=sn*(n-1); /* restore */
X_+=sn*(n-1);

X+=dsn*(n-1); /* jump */
X_+=dsn*(n-1);
sn_=sn+dsn; /* new sn */
sn_=jmpflag;

return pixel[X][Y];
}/** getpixel_ */

int random_(int n)
{
int val;

val=(int)((rand()/(RAND_MAX+1.))*n);

```

```

return val;
}/** random_ **/

long ftell_mem(int i)
{
return fp_mem[i];
}/** ftell_mem **/

void fwrite_mem(int i)
{
rtn[i][fp_mem[i]]=s;
fp_mem[i]++;if(fp_mem[i]>asize-1) refill=0;
}/** fwrite_mem **/

void fread_mem(int i)
{
fp_mem[i]--;if(fp_mem[i]<0) fp_mem[i]=0;
s=rtn[i][fp_mem[i]];
}/** fread_mem **/

int fen(char *str,int i,int jmax)
{
int val;

if(i==jmax+1) val=0;
else if(i==--1) val=jmax;
else val=i;

if(strcmp(str,"X")==0) return enX[val];
else if(strcmp(str,"Y")==0) return enY[val];

else if(strcmp(str,"X_")==0) return enX_[val];
else if(strcmp(str,"Y_")==0) return enY_[val];

else if(strcmp(str,"SN")==0) return enSN[val];
}/** fen **/

void close_vtx(void)
{
int n,i,j,flag,flag_;

```

```

n=RES0;

/* outer and inner(right) */
flag=0; /* 0:open, 1:close */
flag_=0; /* 0:open, 1:close */

i=0;j=0;
if(flag){
/* 1 */
pixel[i*(n-1)+(N1-1)][j*(n-1)+N1-1]=1;
pixel[4*(n-1)][0*(N2-1)]=1;
pixel[4*(n-1)][4*(N2-1)]=1;
/* 2 */
pixel[(i+1)*(n-1)-(N1-1)][j*(n-1)+N1-1]=1;
pixel[4*(n-1)][1*(N2-1)]=1;
/* 3 */
pixel[(i+1)*(n-1)-(N1-1)][(j+1)*(n-1)-(N1-1)]=1;
pixel[4*(n-1)][2*(N2-1)]=1;
/* 4 */
pixel[i*(n-1)+N1-1][(j+1)*(n-1)-(N1-1)]=1;
pixel[4*(n-1)][3*(N2-1)]=1;
}

i=1;
j=0;
if(flag_){
/* 5 */
pixel[i*(n-1)+(N1-1)][j*(n-1)+N1-1]=1;
pixel[5*(n-1)][0*(N2-1)]=1;
pixel[5*(n-1)][4*(N2-1)]=1;
/* 6 */
pixel[(i+1)*(n-1)-(N1-1)][j*(n-1)+N1-1]=1;
pixel[5*(n-1)][1*(N2-1)]=1;
/* 7 */
pixel[(i+1)*(n-1)-(N1-1)][(j+1)*(n-1)-(N1-1)]=1;
pixel[5*(n-1)][2*(N2-1)]=1;
/* 8 */
pixel[i*(n-1)+N1-1][(j+1)*(n-1)-(N1-1)]=1;
pixel[5*(n-1)][3*(N2-1)]=1;
}

j=1;
if(flag_){
/* 9 */
pixel[i*(n-1)+(N1-1)][j*(n-1)+N1-1]=1;

```



```

pixel[6*(n-1)][0*(N2-1)]=1;
pixel[6*(n-1)][4*(N2-1)]=1;
/* 10 */
pixel[(i+1)*(n-1)-(N1-1)][j*(n-1)+N1-1]=1;
pixel[6*(n-1)][1*(N2-1)]=1;
/* 11 */
pixel[(i+1)*(n-1)-(N1-1)][(j+1)*(n-1)-(N1-1)]=1;
pixel[6*(n-1)][2*(N2-1)]=1;
/* 12 */
pixel[i*(n-1)+N1-1][(j+1)*(n-1)-(N1-1)]=1;
pixel[6*(n-1)][3*(N2-1)]=1;
}

```

```

j=2;
if(flag_){
/* 13 */
pixel[i*(n-1)+(N1-1)][j*(n-1)+N1-1]=1;
pixel[7*(n-1)][0*(N2-1)]=1;
pixel[7*(n-1)][4*(N2-1)]=1;
/* 14 */
pixel[(i+1)*(n-1)-(N1-1)][j*(n-1)+N1-1]=1;
pixel[7*(n-1)][1*(N2-1)]=1;
/* 15 */
pixel[(i+1)*(n-1)-(N1-1)][(j+1)*(n-1)-(N1-1)]=1;
pixel[7*(n-1)][2*(N2-1)]=1;
/* 16 */
pixel[i*(n-1)+N1-1][(j+1)*(n-1)-(N1-1)]=1;
pixel[7*(n-1)][3*(N2-1)]=1;
}

```

```

j=3;
if(flag_){
/* 17 */
pixel[i*(n-1)+(N1-1)][j*(n-1)+N1-1]=1;
pixel[8*(n-1)][0*(N2-1)]=1;
pixel[8*(n-1)][4*(N2-1)]=1;
/* 18 */
pixel[(i+1)*(n-1)-(N1-1)][j*(n-1)+N1-1]=1;
pixel[8*(n-1)][1*(N2-1)]=1;
/* 19 */
pixel[(i+1)*(n-1)-(N1-1)][(j+1)*(n-1)-(N1-1)]=1;
pixel[8*(n-1)][2*(N2-1)]=1;
/* 20 */
pixel[i*(n-1)+N1-1][(j+1)*(n-1)-(N1-1)]=1;
pixel[8*(n-1)][3*(N2-1)]=1;
}

```

```

i=2;j=0;
if(flag){
/* 21 */
pixel[i*(n-1)+(N1-1)][j*(n-1)+N1-1]=1;
pixel[9*(n-1)][0*(N2-1)]=1;
pixel[9*(n-1)][4*(N2-1)]=1;
/* 22 */
pixel[(i+1)*(n-1)-(N1-1)][j*(n-1)+N1-1]=1;
pixel[9*(n-1)][1*(N2-1)]=1;
/* 23 */
pixel[(i+1)*(n-1)-(N1-1)][(j+1)*(n-1)-(N1-1)]=1;
pixel[9*(n-1)][2*(N2-1)]=1;
/* 24 */
pixel[i*(n-1)+N1-1][(j+1)*(n-1)-(N1-1)]=1;
pixel[9*(n-1)][3*(N2-1)]=1;
}
}/** close_vtx **/

```

```

int check_v(int x,int y)
{
char AF,af;
int i,m,val;

```

```

/* to af */
AF='A';
for(m=0;m<6;m++){
for(i=0;i<=3;i++){
if(v_(AF,i,x,y)==1){
qx=4+m;qpos=i;
if(qpos==0) val=1;
else if(qpos==1) val=0;
else if(qpos==2) val=3;
else if(qpos==3) val=2;
goto end;
}
}

```

```

AF++;
}

```

```

/* to AF */
af='a';
for(m=0;m<6;m++){
for(i=0;i<=4;i++){

```

```

if(v_(af,i,x,y)==1){
if(m==0)      {qx=0;qy=0;}
else if(m==5) {qx=2;qy=0;}
else          {qx=1;qy=m-1;}

qpos=i%4;
val=4;
goto end;
}
}

af++;
}

/* between inner vertexes(left) */
af='a';
for(m=0;m<6;m++){
for(i=0;i<=4;i++){
if(v_(af,i+5,x,y)==1){
qx=iv[m][i][0][0];
qy=iv[m][i][0][1];
qx_=iv[m][i][1][0];
qy_=iv[m][i][1][1];
val=5;
goto end;
}
}

af++;
}

val=-1;

end:

return val;
}/** check_v **/

int get_dx(int qx,int qy,int qpos)
{
int n,nx,ny;

n=RES0;

if(qx<=3){

```

```

if(qpos==0)      {nx=qx*(n-1)+N1-1;ny=qy*(n-1)+N1-1;}
else if(qpos==1) {nx=(qx+1)*(n-1)-(N1-1);ny=qy*(n-1)+N1-1;}
else if(qpos==2) {nx=(qx+1)*(n-1)-(N1-1);ny=(qy+1)*(n-1)-(N1-1);}
else
      {nx=qx*(n-1)+N1-1;ny=(qy+1)*(n-1)-(N1-1);}
}
else{
nx=qx*(n-1);ny=(qpos%5)*(N2-1);
if(qpos>4) nx-=N1-1;
}

return nx;
}/** get_dx **/

```

```

int get_dy(int qx,int qy,int qpos)
{
int n,nx,ny;

n=RES0;

if(qx<=3){
if(qpos==0)      {nx=qx*(n-1)+N1-1;ny=qy*(n-1)+N1-1;}
else if(qpos==1) {nx=(qx+1)*(n-1)-(N1-1);ny=qy*(n-1)+N1-1;}
else if(qpos==2) {nx=(qx+1)*(n-1)-(N1-1);ny=(qy+1)*(n-1)-(N1-1);}
else
      {nx=qx*(n-1)+N1-1;ny=(qy+1)*(n-1)-(N1-1);}
}
else{
nx=qx*(n-1);ny=(qpos%5)*(N2-1);
if(qpos>4) nx-=N1-1;
}

return ny;
}/** get_dy **/

```

```

void set_vals(int j)
{
int snold;

      enX[j]=X;enY[j]=Y;enX_[j]=X_;enY_[j]=Y_;enSN[j]=sn_;

snold=sn;
sn=/*sn_*/0;

if(pixel[X][Y]==0) getpixel_(X_,Y_,X,Y);
      enX[j]=X;enY[j]=Y;enX_[j]=X_;enY_[j]=Y_;

```

```

sn=snold;
}/** set_vals **/

int cag_r(void)
{
int i,j,dx,dy,n,jmax,dsn;
int flag_[CPMAX],flag_pp[CPMAX],acolor[4*6];
int nx[CPMAX],ny[CPMAX],nx_[CPMAX],ny_[CPMAX],nax[4*6],nay[4*6];
int Nx,Ny,XDP,YDP,x[8],y[8],x_[8],y_[8];
int cp,ssize,posflag,pos,count;
int ca,c1,c2,c3,c4,c5,c7,cflag;
int nxp,nxm,nyp,nym;
int jmp[6+6],sntmp1,sntmp2;

ssize=sizeof(ss);
cp=CPMAX;
n=RESO;

if(N1>=2){
close_vtx();
}
if(CPMAX==3 || CPMAX==6){
putpixel_(1*(n-1),0*(n-1),8);
putpixel_(2*(n-1),2*(n-1),8);
if(N1>=2){
putpixel_(4*(n-1)-(N1-1),1*(N2-1),8);
putpixel_(9*(n-1)-(N1-1),2*(N2-1),8);
}
}
else if(CPMAX==24){
putpixel_(4*(n-1)-(N1-1),0*(N2-1),8);
putpixel_(4*(n-1)-(N1-1),1*(N2-1),8);
putpixel_(4*(n-1)-(N1-1),2*(N2-1),8);
putpixel_(4*(n-1)-(N1-1),3*(N2-1),8);
putpixel_(9*(n-1)-(N1-1),0*(N2-1),8);
putpixel_(9*(n-1)-(N1-1),1*(N2-1),8);
putpixel_(9*(n-1)-(N1-1),2*(N2-1),8);
putpixel_(9*(n-1)-(N1-1),3*(N2-1),8);

for(i=0;i<4;i++){
putpixel_(i*(n-1),0*(n-1),8);
putpixel_(i*(n-1),1*(n-1),8);
}
}
}

```

```
acolor[0]=9;acolor[1]=10;acolor[2]=11;acolor[3]=12;
acolor[4]=1;acolor[5]=2;acolor[6]=3;acolor[7]=4;
acolor[8]=16;acolor[9]=17;acolor[10]=18;acolor[11]=19;
acolor[12]=20;acolor[13]=21;acolor[14]=22;acolor[15]=23;
acolor[16]=24;acolor[17]=25;acolor[18]=26;acolor[19]=27;
acolor[20]=28;acolor[21]=29;acolor[22]=30;acolor[23]=31;
```

```
for(i=0;i<CPMAX;i++){
rcount[i]=0;
flag_[i]=1;
fp_mem[i]=0;
}
```

```
ca=15;
```

```
if(CPMAX==4 && DIV==1){
dx=0;dy=0;
nax[0]=2+dx+DSP      ;nay[0]=2+dy+DSP;
nax[1]=n-3+dx-DSP   ;nay[1]=n-3+dy-DSP;
/*nax[0]=n-3+dx-DSP   ;nay[0]=2+dy+DSP;
nax[1]=2+dx+DSP      ;nay[1]=n-3+dy-DSP;*/
```

```
dx=2*(n-1);dy=0;
if(CROSS==0){ /* // */
nax[2]=n-3+dx-DSP   ;nay[2]=2+dy+DSP;
nax[3]=2+dx+DSP     ;nay[3]=n-3+dy-DSP;
}
else{ /* x */
nax[2]=2+dx+DSP     ;nay[2]=2+dy+DSP;
nax[3]=n-3+dx-DSP   ;nay[3]=n-3+dy-DSP;
}
}
```

```
else if(DIV==0){
dx=0;dy=0;
nax[0]=2+dx+DSP     ;nay[0]=2+dy+DSP;
nax[1]=n-3+dx-DSP   ;nay[1]=2+dy+DSP;
nax[2]=n-3+dx-DSP   ;nay[2]=n-3+dy-DSP;
nax[3]=2+dx+DSP     ;nay[3]=n-3+dy-DSP;
/*nax[0]=2+dx+DSP     ;nay[0]=2+dy+DSP;
nax[1]=n-3+dx-DSP     ;nay[1]=n-3+dy-DSP;*/
```

```
if(CPMAX==3){
/* around s1-v1 */
nax[0]=n-3+dx-DSP   ;nay[0]=2+dy+DSP;
/* around s5-v3 */
```

```

nax[1]=n-1+2+dx+DSP      ;nay[1]=4*(n-1)-2+dy-DSP;
/* around s1-v1 */
nax[2]=n-1+2+dx+DSP      ;nay[2]=2+dy+DSP;
}
}
else if(DIV==1){
/* A */
dx=0;dy=0;
nax[0]=2+dx+DSP          ;nay[0]=2+dy+DSP;
nax[1]=n-3+dx-DSP        ;nay[1]=2+dy+DSP;
nax[2]=n-3+dx-DSP        ;nay[2]=n-3+dy-DSP;
nax[3]=2+dx+DSP          ;nay[3]=n-3+dy-DSP;
/* F */
dx=(0+2)*(n-1);dy=0;
nax[4]=2+dx+DSP          ;nay[4]=2+dy+DSP;
nax[5]=n-3+dx-DSP        ;nay[5]=2+dy+DSP;
nax[6]=n-3+dx-DSP        ;nay[6]=n-3+dy-DSP;
nax[7]=2+dx+DSP          ;nay[7]=n-3+dy-DSP;

/* B */
dx=1*(n-1);dy=0;
nax[8]=2+dx+DSP          ;nay[8]=2+dy+DSP;
nax[9]=n-3+dx-DSP        ;nay[9]=2+dy+DSP;
nax[10]=n-3+dx-DSP       ;nay[10]=n-3+dy-DSP;
nax[11]=2+dx+DSP         ;nay[11]=n-3+dy-DSP;
/* D */
dx=1*(n-1);dy=(0+2)*(n-1);
nax[12]=2+dx+DSP         ;nay[12]=2+dy+DSP;
nax[13]=n-3+dx-DSP       ;nay[13]=2+dy+DSP;
nax[14]=n-3+dx-DSP       ;nay[14]=n-3+dy-DSP;
nax[15]=2+dx+DSP         ;nay[15]=n-3+dy-DSP;

/* C */
dx=1*(n-1);dy=1*(n-1);
nax[16]=2+dx+DSP         ;nay[16]=2+dy+DSP;
nax[17]=n-3+dx-DSP       ;nay[17]=2+dy+DSP;
nax[18]=n-3+dx-DSP       ;nay[18]=n-3+dy-DSP;
nax[19]=2+dx+DSP         ;nay[19]=n-3+dy-DSP;
/* E */
dx=1*(n-1);dy=(1+2)*(n-1);
nax[20]=2+dx+DSP         ;nay[20]=2+dy+DSP;
nax[21]=n-3+dx-DSP       ;nay[21]=2+dy+DSP;
nax[22]=n-3+dx-DSP       ;nay[22]=n-3+dy-DSP;
nax[23]=2+dx+DSP         ;nay[23]=n-3+dy-DSP;

if(CPMAX==6){

```

```

dx=0;dy=0;
/* around s1-v1 */
nax[0]=n-3+dx-DSP      ;nay[0]=2+dy+DSP;
/* around s5-v3 */
nax[1]=n-1+2+dx+DSP    ;nay[1]=4*(n-1)-2+dy-DSP;
/* around s1-v1 */
nax[2]=n-1+2+dx+DSP    ;nay[2]=2+dy+DSP;

/* around s4-v1 */
nax[3]=2*(n-1)-2-DSP;nay[3]=2*(n-1)-2-DSP;
/* around s6-v2 */
nax[4]=3*(n-1)-2-DSP;nay[4]=n-3-DSP;
/* around s4-v1 */
nax[5]=2*(n-1)-2-DSP;nay[5]=2*(n-1)+2+DSP;
}
}

i=0;
while(1){
if(flag_[i]){          /* CP_? */
/*if(i<=2) sn=sn1;else sn=sn2;*/
ig=i;

nx[i]=nax[i];ny[i]=nay[i];
sn=0;
putpixel_(nx[i],ny[i],acolor[i]);
}/**if(flag_[i])**/

i++;if(i==CPMAX) break;
}/**while(1)**/

i=0;
while(1){
if(flag_[i]){          /* CP_? */
nx_[i]=nax[i];ny_[i]=nay[i];
/*if(i<=2) sn=sn1;else sn=sn2;*/
ig=i;

if(combination==0){
}/**if(combination)**/
else{
/* CCW */
if(drn==4) {/* 217 */
if(CPMAX==4 && DIV==1){
if(CROSS==0){ /* // */
if(i==0) {nax[i]++;}

```



```

else if(i==1) {nax[i]--;}
else if(i==2) {nay[i]++;}
else if(i==3) {nay[i]--;}
}
else{ /* x */
    if(i==0) {nax[i]++;}
else if(i==1) {nax[i]--;}
else if(i==2) {nax[i]++;}
else if(i==3) {nax[i]--;}
}
}
else{
#if CPMAX!=3 && CPMAX!=6
    if(i%4==0) {nax[i]++;}
else if(i%4==1) {nay[i]++;}
else if(i%4==2) {nax[i]--;}
else if(i%4==3) {nay[i]--;}
/*    if(i%2==0) {nax[i]++;}
else if(i%2==1) {nax[i]--;}*/
#else
/* around s1-v2 */
    if(i%6==0) {nay[i]--;}
else if(i%6==1) {nay[i]++;}
else if(i%6==2) {nax[i]--;}

/* around s6-v1 */
else if(i%6==3) {nax[i]++;}
/* around s5-v1 */
else if(i%6==4) {nax[i]++;}
else if(i%6==5) {nay[i]--;}
#endif
}
}
}/**else(combination)**/

nx[i]=nax[i];ny[i]=nay[i];

sn=0;
putpixel_(nx[i],ny[i],acolor[i]);
}/**if(flag_[i])**/

i++;if(i==CPMAX) break;
}/**while(1)**/

if(GRPH==1 && cnt==0) use_subroop();
cnt++;

```

```

/***** while(cp) -> *****/

/*999*/
while(cp){
kbhit();
if(refill==0) break;
if(0) use_subroop();
/*printf(" %ld %ld\n",rcount[0],rcount[1]);*/

algo=random_(2);

i=0;
while(1){

if(flag_[i]){ /* CP? */
/*if(Odd!=1) {if(i<=2) sn=sn1;else sn=sn2;}
else sn=sn1;*/
sn=0;
ig=i;

nxp=nx[i]+1;nyp=ny[i]+1;nxm=nx[i]-1;nym=ny[i]-1;
c1=getpixel_(nx[i],ny[i],nxp,ny[i]);
x[1]=X;y[1]=Y;x_[1]=X_;y_[1]=Y_;jmp[1]=sn_;
c2=getpixel_(nx[i],ny[i],nx[i],nyp);
x[2]=X;y[2]=Y;x_[2]=X_;y_[2]=Y_;jmp[2]=sn_;
c3=getpixel_(nx[i],ny[i],nxm,ny[i]);
x[3]=X;y[3]=Y;x_[3]=X_;y_[3]=Y_;jmp[3]=sn_;
c4=getpixel_(nx[i],ny[i],nx[i],nym);
x[4]=X;y[4]=Y;x_[4]=X_;y_[4]=Y_;jmp[4]=sn_;

Nx=nx[i];Ny=ny[i];

if(sn==0) dsn=4;
else if(sn==4 && Ny>3*(n-1)) dsn=-4;
else if(sn==4 && Ny<3*(n-1)) dsn=4;
else if(sn==8) dsn=-4;

posflag=check_v(Nx,Ny);

if(posflag==-1){
if((c1==ca)||(c2==ca)||(c3==ca)||(c4==ca)) cflag=1;
else cflag=0;
}
else if(posflag==0){

```

```

if((c1==ca)|| (c2==ca)|| (c3==ca)|| (c4==ca)) cflag=1;
else{
j=4;rot(0,2);X_=get_dx(qx,qy,qpos);Y_=get_dy(qx,qy,qpos);X=X_+tmp0;Y=Y_+tmp1;
    enX[j]=X;enY[j]=Y;enX_[j]=X_;enY_[j]=Y_;enSN[j]=sn_;

cflag=0;
for(j=4;j<=4;j++) {if(pixel[enX[j]][enY[j]]==ca) {cflag=1;break;}}
}
}
else if(posflag==1){
if((c1==ca)|| (c2==ca)|| (c3==ca)|| (c4==ca)) cflag=1;
else{
j=4;rot(0,2);X_=get_dx(qx,qy,qpos);Y_=get_dy(qx,qy,qpos);X=X_+tmp0;Y=Y_+tmp1;
    enX[j]=X;enY[j]=Y;enX_[j]=X_;enY_[j]=Y_;enSN[j]=sn_;

cflag=0;
for(j=4;j<=4;j++) {if(pixel[enX[j]][enY[j]]==ca) {cflag=1;break;}}
}
}
else if(posflag==2){
if((c1==ca)|| (c2==ca)|| (c3==ca)|| (c4==ca)) cflag=1;
else{
j=4;rot(0,2);X_=get_dx(qx,qy,qpos);Y_=get_dy(qx,qy,qpos);X=X_+tmp0;Y=Y_+tmp1;
    enX[j]=X;enY[j]=Y;enX_[j]=X_;enY_[j]=Y_;enSN[j]=sn_;

cflag=0;
for(j=4;j<=4;j++) {if(pixel[enX[j]][enY[j]]==ca) {cflag=1;break;}}
}
}
else if(posflag==3){
if((c1==ca)|| (c2==ca)|| (c3==ca)|| (c4==ca)) cflag=1;
else{
j=4;rot(0,2);X_=get_dx(qx,qy,qpos);Y_=get_dy(qx,qy,qpos);X=X_+tmp0;Y=Y_+tmp1;
    enX[j]=X;enY[j]=Y;enX_[j]=X_;enY_[j]=Y_;enSN[j]=sn_;

cflag=0;
for(j=4;j<=4;j++) {if(pixel[enX[j]][enY[j]]==ca) {cflag=1;break;}}
}
}
else if(posflag==4){
if((c1==ca)|| (c2==ca)|| (c3==ca)|| (c4==ca)) cflag=1;
else{
j=3;rot(qpos,-1);X_=get_dx(qx,qy,qpos);Y_=get_dy(qx,qy,qpos);X=X_+tmp0;Y=Y_+tmp1;
    enX[j]=X;enY[j]=Y;enX_[j]=X_;enY_[j]=Y_;enSN[j]=sn_;
j=4;rot(qpos,-2);X_=get_dx(qx,qy,qpos);Y_=get_dy(qx,qy,qpos);X=X_+tmp0;Y=Y_+tmp1;
    enX[j]=X;enY[j]=Y;enX_[j]=X_;enY_[j]=Y_;enSN[j]=sn_;
}
}
}
}
}

```

```

cflag=0;
for(j=3;j<=4;j++) {if(pixel[enX[j]][enY[j]]==ca) {cflag=1;break;}}
}
}
else if(posflag==5){
if((c1==ca)||(c2==ca)||(c3==ca)||(c4==ca)) cflag=1;
else{
j=3;rot(1,-1);X_=qx;Y_=qy;X=X_+tmp0;Y=Y_+tmp1;
set_vals(j);
j=4;rot(1,-2);X_=qx;Y_=qy;X=X_+tmp0;Y=Y_+tmp1;
set_vals(j);
j=5;rot(1,-1);X_=qx_;Y_=qy_;X=X_+tmp0;Y=Y_+tmp1;
set_vals(j);

cflag=0;
for(j=3;j<=5;j++) {if(pixel[enX[j]][enY[j]]==ca) {cflag=1;break;}}
}
}

if(cflag){
s.xx=nx[i];s.yy=ny[i];s.xx_=nx_[i];s.yy_=ny_[i];s.sn=sn;fwrite_mem(i);

std_x=nx[i];std_y=ny[i]; /* S */
last_x=nx_[i];last_y=ny_[i];
/*nx_[i]=nx[i];ny_[i]=ny[i];*/ /* new b */

/* for out */
if(sn==0) dsn=4;
else if(sn==4 && Ny>3*(n-1)) dsn=-4;
else if(sn==4 && Ny<3*(n-1)) dsn=4;
else if(sn==8) dsn=-4;

posflag=check_v(Nx,Ny);

if(posflag==-1){
j=-1;

if(c1!=0){
j++; /* ca1 */
enX[j]=x[1];enY[j]=y[1];enX_[j]=x_[1];enY_[j]=y_[1];enSN[j]=jmp[1];
}
if(c4!=0){
j++; /* ca4 */

```

```

    enX[j]=x[4];enY[j]=y[4];enX_[j]=x_[4];enY_[j]=y_[4];enSN[j]=jmp[4];
}
if(c3!=0){
j++; /* ca3 */
    enX[j]=x[3];enY[j]=y[3];enX_[j]=x_[3];enY_[j]=y_[3];enSN[j]=jmp[3];
}
if(c2!=0){
j++; /* ca2 */
    enX[j]=x[2];enY[j]=y[2];enX_[j]=x_[2];enY_[j]=y_[2];enSN[j]=jmp[2];
}
}
else if(posflag==0){
/* v1 */
j=0; /* ca2 */
    enX[j]=x[2];enY[j]=y[2];enX_[j]=x_[2];enY_[j]=y_[2];enSN[j]=jmp[2];
j=1; /* ca1 */
    enX[j]=x[1];enY[j]=y[1];enX_[j]=x_[1];enY_[j]=y_[1];enSN[j]=jmp[1];
j=2; /* ca4 */
    enX[j]=x[4];enY[j]=y[4];enX_[j]=x_[4];enY_[j]=y_[4];enSN[j]=jmp[4];
j=3; /* ca3 */
    enX[j]=x[3];enY[j]=y[3];enX_[j]=x_[3];enY_[j]=y_[3];enSN[j]=jmp[3];

        /* out */
j=4;rot(0,2);X_=get_dx(qx,qy,qpos);Y_=get_dy(qx,qy,qpos);X=X_+tmp0;Y=Y_+tmp1;
    enX[j]=X;enY[j]=Y;enX_[j]=X_;enY_[j]=Y_;enSN[j]=sn_;
}
else if(posflag==1){
/* v0 */
j=0; /* ca1 */
    enX[j]=x[1];enY[j]=y[1];enX_[j]=x_[1];enY_[j]=y_[1];enSN[j]=jmp[1];
j=1; /* ca4 */
    enX[j]=x[4];enY[j]=y[4];enX_[j]=x_[4];enY_[j]=y_[4];enSN[j]=jmp[4];
j=2; /* ca3 */
    enX[j]=x[3];enY[j]=y[3];enX_[j]=x_[3];enY_[j]=y_[3];enSN[j]=jmp[3];
j=3; /* ca2 */
    enX[j]=x[2];enY[j]=y[2];enX_[j]=x_[2];enY_[j]=y_[2];enSN[j]=jmp[2];

        /* out */
j=4;rot(0,2);X_=get_dx(qx,qy,qpos);Y_=get_dy(qx,qy,qpos);X=X_+tmp0;Y=Y_+tmp1;
    enX[j]=X;enY[j]=Y;enX_[j]=X_;enY_[j]=Y_;enSN[j]=sn_;
}
else if(posflag==2){
/* v3 */
j=0; /* ca4 */
    enX[j]=x[4];enY[j]=y[4];enX_[j]=x_[4];enY_[j]=y_[4];enSN[j]=jmp[4];
j=1; /* ca3 */

```

```

    enX[j]=x[3];enY[j]=y[3];enX_[j]=x_[3];enY_[j]=y_[3];enSN[j]=jmp[3];
j=2; /* ca2 */
    enX[j]=x[2];enY[j]=y[2];enX_[j]=x_[2];enY_[j]=y_[2];enSN[j]=jmp[2];
j=3; /* ca1 */
    enX[j]=x[1];enY[j]=y[1];enX_[j]=x_[1];enY_[j]=y_[1];enSN[j]=jmp[1];

        /* out */
j=4;rot(0,2);X_=get_dx(qx,qy,qpos);Y_=get_dy(qx,qy,qpos);X=X_+tmp0;Y=Y_+tmp1;
    enX[j]=X;enY[j]=Y;enX_[j]=X_;enY_[j]=Y_;enSN[j]=sn_;
}
else if(posflag==3){
/* v2 */
j=0; /* ca3 */
    enX[j]=x[3];enY[j]=y[3];enX_[j]=x_[3];enY_[j]=y_[3];enSN[j]=jmp[3];
j=1; /* ca2 */
    enX[j]=x[2];enY[j]=y[2];enX_[j]=x_[2];enY_[j]=y_[2];enSN[j]=jmp[2];
j=2; /* ca1 */
    enX[j]=x[1];enY[j]=y[1];enX_[j]=x_[1];enY_[j]=y_[1];enSN[j]=jmp[1];
j=3; /* ca4 */
    enX[j]=x[4];enY[j]=y[4];enX_[j]=x_[4];enY_[j]=y_[4];enSN[j]=jmp[4];

        /* out */
j=4;rot(0,2);X_=get_dx(qx,qy,qpos);Y_=get_dy(qx,qy,qpos);X=X_+tmp0;Y=Y_+tmp1;
    enX[j]=X;enY[j]=Y;enX_[j]=X_;enY_[j]=Y_;enSN[j]=sn_;
}
else if(posflag==4){
/* v3(right) */
j=0; /* ca4 */
    enX[j]=x[4];enY[j]=y[4];enX_[j]=x_[4];enY_[j]=y_[4];enSN[j]=jmp[4];
j=1; /* ca3 */
    enX[j]=x[3];enY[j]=y[3];enX_[j]=x_[3];enY_[j]=y_[3];enSN[j]=jmp[3];
j=2; /* ca2 */
    enX[j]=x[2];enY[j]=y[2];enX_[j]=x_[2];enY_[j]=y_[2];enSN[j]=jmp[2];

        /* out */
j=3;rot(qpos,-1);X_=get_dx(qx,qy,qpos);Y_=get_dy(qx,qy,qpos);X=X_+tmp0;Y=Y_+tmp1;
    enX[j]=X;enY[j]=Y;enX_[j]=X_;enY_[j]=Y_;enSN[j]=sn_;
j=4;rot(qpos,-2);X_=get_dx(qx,qy,qpos);Y_=get_dy(qx,qy,qpos);X=X_+tmp0;Y=Y_+tmp1;
    enX[j]=X;enY[j]=Y;enX_[j]=X_;enY_[j]=Y_;enSN[j]=sn_;
}
else if(posflag==5){
/* v1(left) */
j=0; /* ca2 */
    enX[j]=x[2];enY[j]=y[2];enX_[j]=x_[2];enY_[j]=y_[2];enSN[j]=jmp[2];
j=1; /* ca1 */
    enX[j]=x[1];enY[j]=y[1];enX_[j]=x_[1];enY_[j]=y_[1];enSN[j]=jmp[1];

```

```

j=2; /* ca4 */
    enX[j]=x[4];enY[j]=y[4];enX_[j]=x_[4];enY_[j]=y_[4];enSN[j]=jmp[4];

                /* out */
j=3;rot(1,-1);X_=qx;Y_=qy;X=X_+tmp0;Y=Y_+tmp1;
set_vals(j);
j=4;rot(1,-2);X_=qx;Y_=qy;X=X_+tmp0;Y=Y_+tmp1;
set_vals(j);
j=5;rot(1,-1);X_=qx_;Y_=qy_;X=X_+tmp0;Y=Y_+tmp1;
set_vals(j);
}

jmax=j;

for(j=0;j<=jmax;j++){
if(enX[j]==nx_[i] && enY[j]==ny_[i]) {pos=j;break;}
}

if(algo==0){
/* CW */
count=0;
for(j=pos;;){
if(pixel[fen("X",j,jmax)][fen("Y",j,jmax)]!=ca &&
    pixel[fen("X",j-1,jmax)][fen("Y",j-1,jmax)]==ca){
nx[i]=fen("X",j-1,jmax);ny[i]=fen("Y",j-1,jmax);
nx_[i]=fen("X_",j-1,jmax);ny_[i]=fen("Y_",j-1,jmax);
if(i<=2) sntmp1=fen("SN",j-1,jmax);
else sntmp2=fen("SN",j-1,jmax);
break;
}

j--;if(j<0) j=jmax;
count++;if(count==jmax+1) {printf(" ?CW\n");break;}
}/**for()**/
}
else{
/* CCW */
count=0;
for(j=pos;;){
if(pixel[fen("X",j,jmax)][fen("Y",j,jmax)]!=ca &&
    pixel[fen("X",j+1,jmax)][fen("Y",j+1,jmax)]==ca){
nx[i]=fen("X",j+1,jmax);ny[i]=fen("Y",j+1,jmax);
nx_[i]=fen("X_",j+1,jmax);ny_[i]=fen("Y_",j+1,jmax);
if(i<=2) sntmp1=fen("SN",j+1,jmax);
else sntmp2=fen("SN",j+1,jmax);
break;
}
}
}

```

```

}

j++;if(j>jmax) j=0;
count++;if(count==jmax+1) {printf(" ?CCW\n");break;}
}/**for()**/
}

/*if(sntmp1>=113) printf(" jump\n");*/
/*printf(" %ld\n",rcount[i]);*/

if(1){
/*if(i<2){
sn=sntmp1;
}
else if(i==2){
sn=sntmp1;
sn1=sntmp1;
}
else if(i>2 && i<5){
sn=sntmp2;
}
else if(i==5){
sn=sntmp2;
sn2=sntmp2;
}*/

sn=0;

/*if(rcount[0]==23 && i==0) c1=3;
else if(rcount[0]==24 && i==0) c1=4;
else if(rcount[0]==25 && i==0) c1=5;
else if(rcount[0]==26 && i==0) c1=6;
else if(rcount[1]==23 && i==1) c1=3+8;
else if(rcount[1]==24 && i==1) c1=4+8;
else if(rcount[1]==25 && i==1) c1=5+8;
else if(rcount[1]==26 && i==1) c1=6+8;
else c1=acolor[i];

if(rcount[0]>=23 && rcount[0]<=26)
printf(" i:%d S:%d %d S_:%d %d\n",i,nx[i],ny[i],nx_[i],ny_[i]);
putpixel_(nx[i],ny[i],c1);*/

putpixel_(nx[i],ny[i],acolor[i]);
}

flag_pp[i]=1;

```



```
 }/**if(c1,c2,c3,c4)**/  
 else{  
 if(ftell_mem(i)==0) {flag_[i]=0;cp--;if(cp==0) break;}  
 fread_mem(i);  
 nx[i]=s.xx;ny[i]=s.yy;nx_[i]=s.xx_;ny_[i]=s.yy_;  
 /*if(i<=2) sn1=s.sn;  
 else sn2=s.sn;*/  
 flag_pp[i]=0;  
 }/**else(c1,c2,c3,c4)**/  
 }/**if(flag_[i])**/  
  
 i++;if(i==CPMAX) break;  
 }/**while(1)**/  
  
 if(0){  
 }  
 }/**while(cp)**/  
  
 /*putpixel(3*(n-1),4*(n-1),0);  
 putpixel((3+4)*(n-1),4*(n-1),0);*/  
  
 return 0;  
 }/** cag_r **/
```