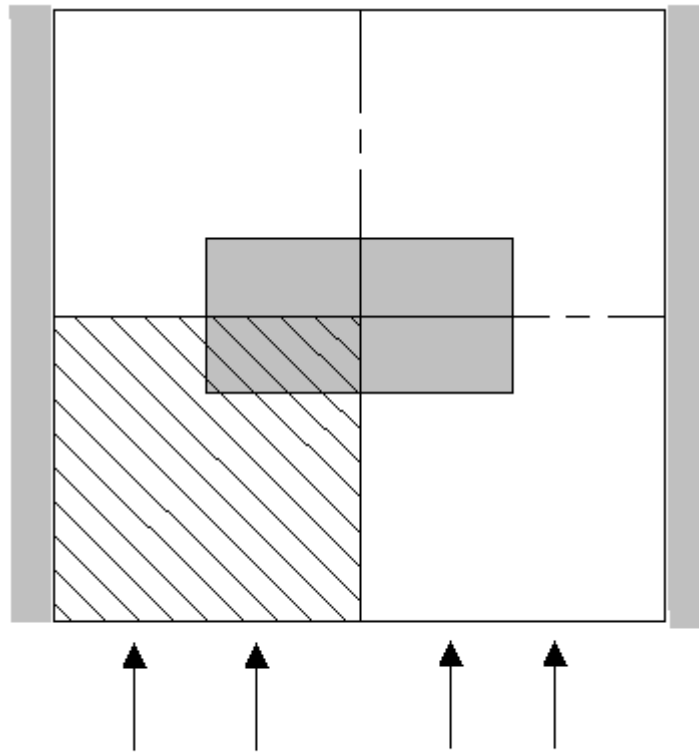
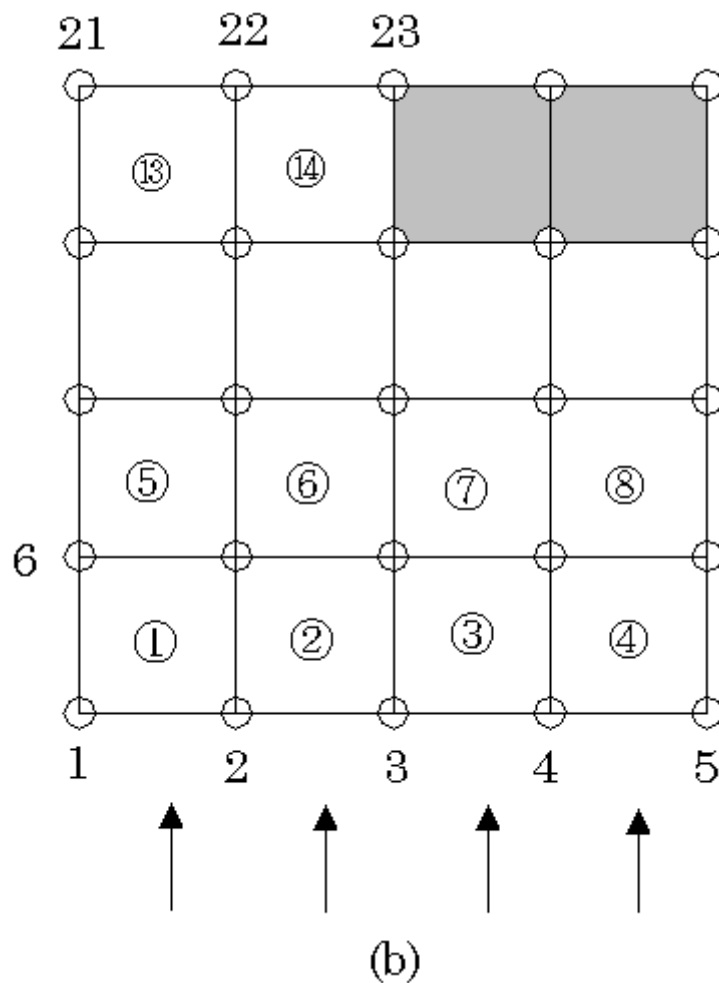


2次元ポテンシャル流れのポテンシャル等高線



(a)



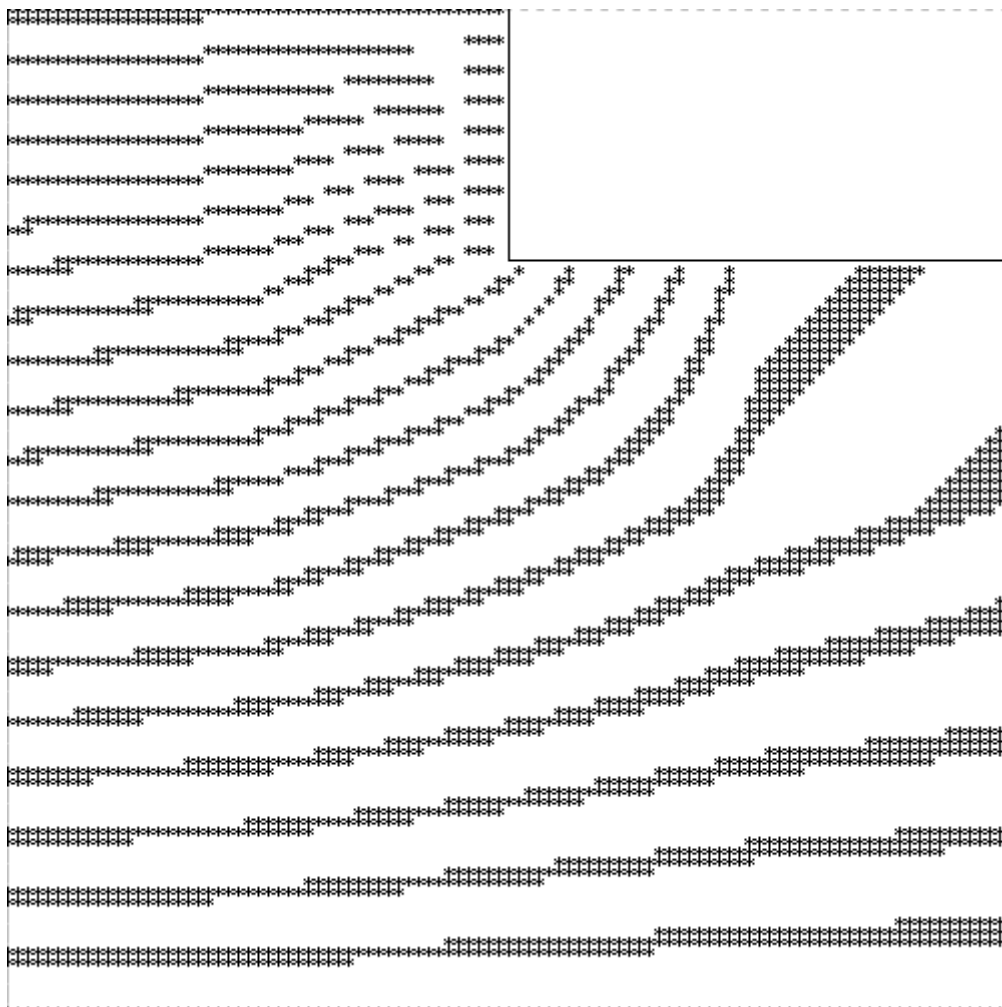
四角の障害物のある流れのポテンシャル等高線をもとめてみよう。上図の (a) のように、左右の壁の間にはさまれて上方向へゆっくり流れるポテンシャル流れがあり、中央に灰色でしめした四角の障害物があるものとしよう。対称性から、下左の斜線をほどこした $1/4$ の領域について、(b) のように、14 個の正方形のメッシュにわけ、1 から 2 3 までの節点番号をつける。計算のための、各要素にたいする 4 個の節点の順番は左下からはじまり、反時計周りに数えるものとする。例えば要素 7 では 1, 2, 7, 6 という具合に。

(1) 各要素番号と対応する 4 個の節点番号の関係を示すトポロジーテーブルを作成する。

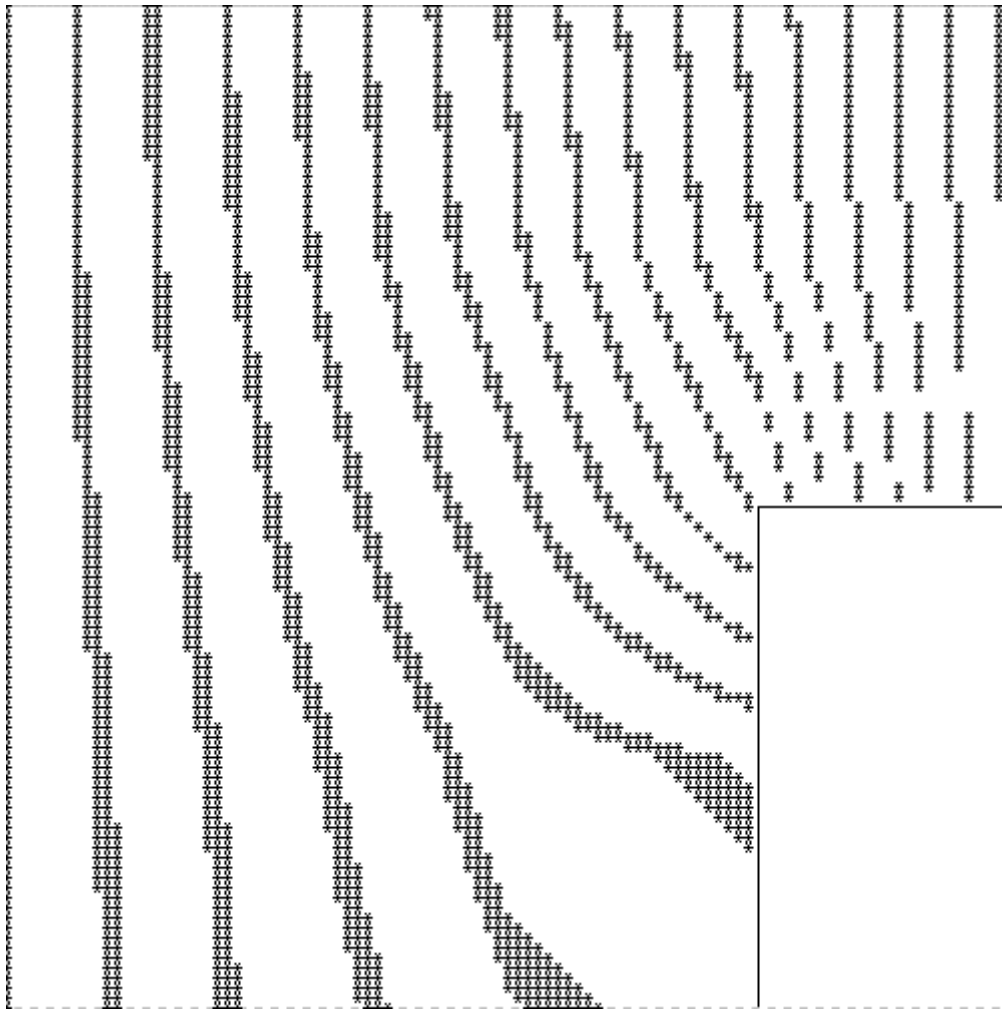
(2) 境界条件として、黒点をつけた節点の流れ関数の値を 0 とする。また節点 1 . . . 5 のポテンシャル関数の値を 4 とする。節点 2 1, 2 2, 2 3 のポテンシャル関数の値を 0 とする。

- (3) ラプラスの微分方程式を数値計算するための要素剛性マトリクスを作成する。
- (4) トポロジーテーブルを利用して全体剛性マトリクスを作成する。
- (5) 境界条件によって全体剛性マトリクスと荷重ベクトルを修正する。
- (6) 線形方程式を解く。
- (7) 求めた流れの関数の各節点の値をもとにして、ポテンシャル等高線を描く。

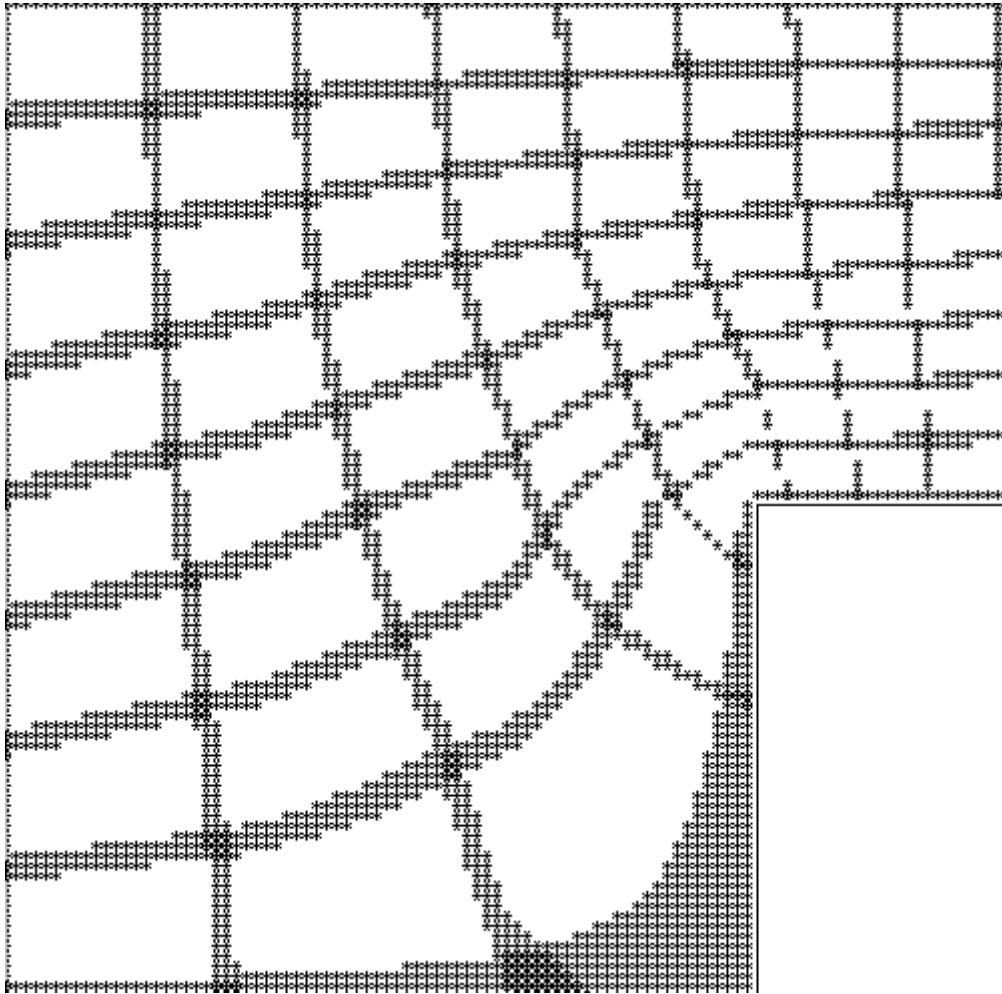
ポテンシャル等高線は次のようになる。



流線の数値計算をやった場合には、障害物は右下にあり、流れは左 右に流れたので、上の図を-90度回転させてみよう。



線の間隔を広げて、流線のグラフと -90 度回転させたポテンシャル等高線の図を重ね合わせてみよう。



少々不正確なところもあるが、まあこんなところで我慢するでしょうか。

! Finite Element Method -Velocity Potential

!fem101.bas by T. Uede 00-8-12

set window 0,100,0,100

draw grid with scale (100,100)

option base 0

dim u(25),Kel(16,4,4),Top(16,4),Res(25,2),K(25,25),Kmod(25,25)

dim fmod(25),a(25,25),c(25)

for i=1 to 4 !-----topology

for j=1 to 4

```

    let Top(i+4*(j-1),1)=i+4*(j-1)+j-1
    let Top(i+4*(j-1),2)=i+4*(j-1)+j
    let Top(i+4*(j-1),3)=i+4*(j-1)+j+5
    let Top(i+4*(j-1),4)=i+4*(j-1)+j+4
  next j
next i

for i=1 to 5 !-----boundary condition
  let Res(i,1)=i
  let Res(i,2)=4
next i
for i=1 to 3
  let Res(i+5,1)=20+i
  let Res(i+5,2)=0
next i

for ie=1 to 14 !-----element stiffness matrix
  let Kel(ie,1,1)=2/6+2/6
  let Kel(ie,1,2)=-2/6+1/6
  let Kel(ie,1,3)=-1/6-1/6
  let Kel(ie,1,4)=1/6-2/6

  let Kel(ie,2,1)=Kel(ie,1,2)
  let Kel(ie,2,2)=2/6+2/6
  let Kel(ie,2,3)=1/6-2/6
  let Kel(ie,2,4)=-1/6-1/6

  let Kel(ie,3,1)=Kel(ie,1,3)
  let Kel(ie,3,2)=Kel(ie,2,3)
  let Kel(ie,3,3)=2/6+2/6
  let Kel(ie,3,4)=-2/6+1/6

  let Kel(ie,4,1)=Kel(ie,1,4)
  let Kel(ie,4,2)=Kel(ie,2,4)
  let Kel(ie,4,3)=Kel(ie,3,4)
  let Kel(ie,4,4)=2/6+2/6

```

```

next ie

for ie=1 to 14 !-----global stiffness matrix
  for i=1 to 4
    for j=1 to 4
      let K(Top(ie,i),Top(ie,j))=K(Top(ie,i),Top(ie,j))+Kel(ie,i,j)
    next j
  next i
next ie

for i=1 to 23 !-----modifying the matrix
  for j=1 to 23
    let Kmod(i,j)=K(i,j)
  next j
next i
for i=1 to 23
  for j=1 to 23
    for ires=1 to 8
      if i=Res(ires,1) then
        let Kmod(i,j)=0
        let Kmod(j,i)=0
        let Kmod(i,i)=1
      end if
    next ires
  next j
next i
for i=1 to 23 !-----modifying the force vector
  let fmod(i)=0
next i
for i=1 to 23
  for ires=1 to 8
    let fmod(i)=fmod(i)-K(i,Res(ires,1))*Res(ires,2)
    let fmod(Res(ires,1))=Res(ires,2)
  next ires
next i
for i=1 to 23 !-----solving the linear equations

```

```

let c(i)=fmod(i)
for j=1 to 23
  let a(i,j)=Kmod(i,j)
next j
next i
for j=1 to 23
  for i=j+1 to 23
    let pp=a(i,j)/a(j,j)
    let c(i)=c(i)-c(j)*pp
    for ind=j to 22
      let a(i,ind)=a(i,ind)-a(j,ind)*pp
    next ind
  next i
next j
let u(23)=c(23)/a(23,23)
for j=22 to 1 step -1
  let s=0
  for i=j+1 to 22
    let s=s+a(j,i)*u(i)
  next i
  let u(j)=(c(j)-s)/a(j,j)
next j

for x=0 to 100 !-----drawing the flow line
  for y=0 to 100
    for i=1 to 4
      for j=1 to 4
        if x>=25*(i-1) and x<=25*i then
          if y>=25*(j-1) and y<=25*j then
            if x<50 or y<75 then
              let xx=x-25*(i-1)
              let yy=y-25*(j-1)
              let ie=i+4*(j-1)
              let uu=(1-xx/25)*(1-yy/25)*u(Top(ie,1))
              let uu=uu+xx/25*(1-yy/25)*u(Top(ie,2))
              let uu=uu+xx*yy/25/25*u(Top(ie,3))

```



```

        let uu=uu+(1-xx/25)*yy/25*u(Top(ie,4))
      end if
    end if
  end if
next j
next i

for i=1 to 5 step 0.5
  if uu>=i-1 and uu<i-1+0.05 then
    if x<50 or y<75 then
      plot points: x,y
    end if
  end if
next i
let uu=0
next y
next x
plot lines :50,100;50,75;100,75
END

```