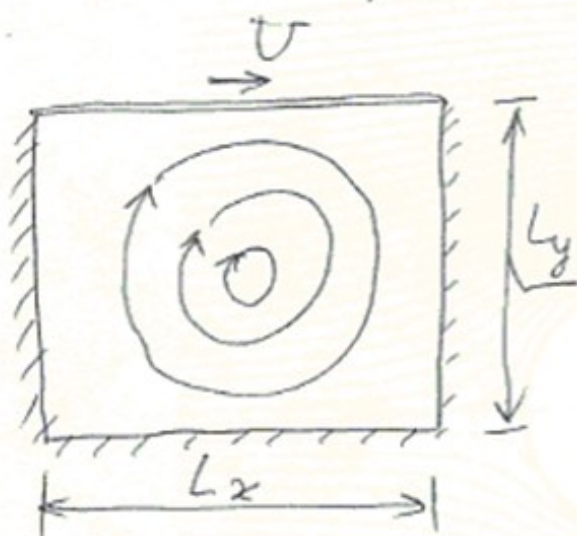


Лаб. работа № 5 "Ламинарное течение в канале"



ЗАДАНИЕ С помощью

МКР и МКО
провести расчеты
ламинарного
течения несжима-
емой жидкости
в прямоуг. канале

при числах Рейнольдса $Re = \frac{UL_x}{\nu} = 10^2 \text{ и } 10^3$

Принять $L_{\max} = \max\{L_x, L_y\} = 1 \text{ м}$

Основное данное приведено в табл.:

№ вар.	1	2	3	4	5	6	7
жидкость	вода	масло	вода	спирт	азот	вода	керосин
L_y/L_x	2/3	3/2	1/2	2	3/5	5/3	3/7
№ вар.	8	9	10	11	12	13	14
жидкость	метан	вода	кислород	масло	керосин	спирт	вода
L_y/L_x	7/3	3/4	4/3	4/5	5/4	2/5	5/2
№ вар.	15	16	17	18	19	20	21
жидкость	азот	метан	кислород	спирт	масло	керосин	этан
L_y/L_x	5/6	6/5	5/7	7/5	5/8	8/5	5/9

УКАЗАНИЯ

МКР

- 1) Внести необх. изменения в текущую программу. Написать явно процедуры "UVCalc" и "DIVCalc", в к-рых рассчитать векторное поле скоростей

$$u = \frac{\partial \psi}{\partial y}, \quad v = -\frac{\partial \psi}{\partial x}$$

и его ^{максимальная} дивергенция

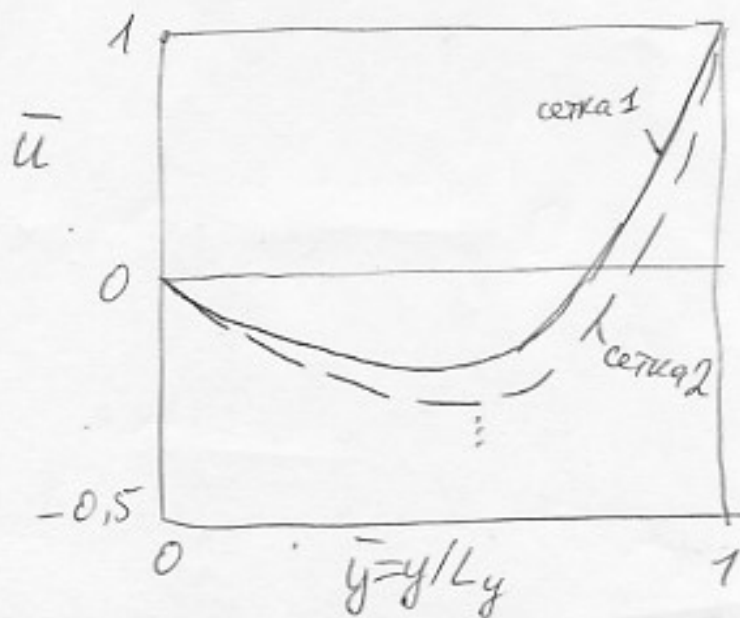
$$(\operatorname{div} \vec{V})_{\max} = \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right)_{\max}$$

Для контроля правильности проведения расчёта для квадр. каверны.

- 2) Для случая $Re = 10^2$ убедиться в наличии сходимость по сетке

(при $\bar{h}_x = \bar{h}_y = h/L_x = \frac{1}{20}, \frac{1}{40}, \frac{1}{80}$)

профиль относительной продольной скорости $\bar{u} = u/U$ в среднем верт. сечении каверны $x = L_x/2$.



Дальнейшие расчеты проводить на сетке, обеспечившей са-еб ре-тов.

МКО

- 1) Сетка - структурированная с помощью "Face meshing", число ячеек - по выбранной конечно-разностной сетке.
- 2) Решатель (Fluent) - настройки по умолчанию, за исключением "Materials" (выбрать из базис данных) и порядка точности схемы ("First" \rightarrow "Second")

СОДЕРЖАНИЕ ОТЧЕТА

- 1) Подготовка задачи: рисунок вашей каверы и наглядно - краевая задача для завихренности ω и функции тока ψ .

- 2) В МКО

а) Текст Фортран-программы

б) Исследование ск-сти по сетке

в) Ре-тг для $Re = 10^2$ и 10^3 на выбранной в и. б) сетке:

- графики установившейся завихренности

$$\lg \omega = f(t)$$

— векторное поле скоростей

3) В МК0

а) Процессинг:

- растровая сетка
- снимки экрана с внесенными во Fluent настройками

б) Результаты для $Re = 10^2$ и 10^3 :

- траектории невязок
- векторное поле скоростей
- картина линий тока

4) Сравнение МКР и МК0

- профили $\bar{u}(\bar{x} = \frac{1}{2}, \bar{y})$ и $\bar{v}(\bar{x} = \frac{1}{2}, \bar{y})$

```

program lab_cav1 ! Cavity steady flow
use GrafLib
implicit real*8(a-h,o-z)
parameter(Nx=81,Ny=81)
dimension x(Nx,Ny),y(Nx,Ny)
dimension dzeta(Nx,Ny),psi(Nx,Ny),u(Nx,Ny),v(Nx,Ny)
open(10,file='cav1.dat')
open(11,file='cav1.txt')
anu=1.e-6; Uw=1e-3; aLx=1.0; aLy=1.; Re=Uw*aLx/anu
aLmax=dmax1(aLx,aLy)
dx=aLx/(Nx-1); dy=aLy/(Ny-1)
dx1=2*dx; dy1=2*dy
dx2=dx**2; dy2=dy**2; dxy2=2*(dx2+dy2)
si1=dy2/dxy2; si2=dx2/dxy2; si3=dx2*dy2/dxy2
dt1=1./(2*anu*(1/dx2+1/dy2))
Nt=100000; nout=1000
itmax=10; eps=1d-8; omega=1.5d0
write(*,*) '*** Cavity Flow ***';write(10,*) '*** Cavity Flow ***'
write(*, '(a10,f6.3,a2,f6.3,a2)') 'Geometry:',aLx,' x',aLy,' m'
write(10,'(a10,f6.3,a2,f6.3,a2)') 'Geometry:',aLx,' x',aLy,' m'
write(*,*) 'Re=',Re
write(*,*) 'Mesh:',Nx,' x',Ny,' nodes';write(10,*) 'Mesh:',Nx,' x',Ny,' nodes'
do i=1,Nx; do j=1,Ny
  x(i,j)=(i-1)*dx; y(i,j)=(j-1)*dy
end do; end do
t=0; psi=0; dzeta=0
u=0; u(:,Ny)=Uw; v=0
V2max=maxval(u**2+v**2)
if(V2max<1e-6)then
  dt2=1e6
else
  dt2=2*anu/V2max
end if
dt=min(dt1,dt2)
write(*,*) ' dt=',dt
write(10,*) ' dt=',dt
tmax=Nt*dt

xgmin=0.0d0; xgmax=1.0d0 !dble(tmax)
ygmin=-10.0d0; ygmax=0.0d0
call GrafInit(1,'Cavity: Max. Rate of Vorticity')
ii=clickmenuqq(QWIN$TILE)
call Axis('t/tmax','lg(ddzeta)_max',0,2d-3)
call GRID(5,10,0)

ddzeta=1.0
write(*,*) ' n      t      d(dzeta) it'
write(10,*) ' n      t      d(dzeta) it'
do n=1,Nt

```

```

told=t; ddzetaold=ddzeta
call DzetaSol(Gmax)
t=t+dt
ddzeta=Gmax*dt
call PsiSol(it)
call UVCalc()
ig=setactiveqq(1)
call Plot2(dble((/told/tmax,t/tmax/)),dble((/dlog10(ddzetaold),dlog10(ddzeta)/)),3,2,2d-3)
if(mod(n,nout)==0) then
  write(*,100)n,t,ddzeta,it
  write(10,100)n,t,ddzeta,it
end if
if(ddzeta<1e-9)then
  write(*,100)n,t,ddzeta,it
  write(10,100)n,t,ddzeta,it
  exit
end if
end do

```

```

psimin=minval(psi); dzetamin=minval(dzeta)
umin=minval(u); vmin=minval(v)
write(*,*)'Flow Statistics:';write(10,*)'Flow Statistics:'
write(*,*)' psimin=',psimin,' dzetamin=',dzetamin,' umin=',umin,' vmin=',vmin
write(10,*)' psimin=',psimin,' dzetamin=',dzetamin,' umin=',umin,' vmin=',vmin
psimax=maxval(psi); dzetamax=maxval(dzeta)
umax=maxval(u); vmax=maxval(v)
write(*,*)' psimax=',psimax,' dzetamax=',dzetamax,' umax=',umax,' vmax=',vmax
write(10,*)' psimax=',psimax,' dzetamax=',dzetamax,' umax=',umax,' vmax=',vmax
pause

```

```

call DIVCalc(divmax)
write(*,*)' div(V)_max=',divmax
write(10,*)' div(V)_max=',divmax
ii=SAVEIMAGE_W('rate.bmp',xgmin-0.1*(xgmax-xgmin),ygmax+0.1*(ygmax-ygmin) &
               ,xgmax+0.1*(xgmax-xgmin),ygmin-0.1*(ygmax-ygmin))
xgmin=0.; xgmax=aLmax !1.
ygmin=0.; ygmax=aLmax !xgmax
call GrafInit(2,'Cavity: Velocity Vectors')
ii=clickmenuqq(QWIN$TILE)
ig=setactiveqq(2)
call Plot2((/0.0d0,aLx,aLx,0.0d0,0.0d0/),(/0.0d0,0.0d0,aLy,aLy,0.0d0/),0,1000,2d-3)
!call Vectors(Nx,Ny,x,y,u,v,1,2,1)
call Vectors1(Nx,Ny,x(:,1),y(1,:),u,v,0.04d0,1,3,1d-3,1)
ii=SAVEIMAGE_W('vectors.bmp',xgmin,ygmax,xgmax,ygmin)
xgmin=0.0d0; xgmax=1.0d0
ygmin=-0.4d0; ygmax=1.0d0
call GrafInit(3,'Cavity: Velocity u & v-components at x=Lx/2')
ii=clickmenuqq(QWIN$TILE)
ig=setactiveqq(3)

```

```

call Axis('y/Ly',' u/Uw, v/Uw at x=Lx/2',0,1d-3)
call GRID(6,8,0)
call Plot2(y(1,:)/aLy,u(Nx/2+1,:)/Uw,1,200,3d-3)
call Plot2(y(1,:)/aLy,v(Nx/2+1,:)/Uw,2,200,3d-3)
ii=SAVEIMAGE_W('uv.bmp',xgmin-0.1,ygmax+0.1,xgmax+0.1,ygmin-0.1)

```

```

write(11,*)Nx,Ny
write(11,'(21(1x,1pe10.3))')x(:,1); write(11,*)' '
write(11,'(21(1x,1pe10.3))')y(1,:); write(11,*)' '
write(11,'(21(1x,1pe10.3))')(u(:,j)/Uw,j=1,Ny); write(11,*)' '
write(11,'(21(1x,1pe10.3))')(v(:,j)/Uw,j=1,Ny); write(11,*)' '
write(11,'(21(1x,1pe10.3))')(psi(:,j)/psimax,j=1,Ny); write(11,*)' '
write(11,'(21(1x,1pe10.3))')(dzeta(:,j)/dzetamax,j=1,Ny)
100 format(i5,2(1x,1pe11.4),i3)
pause

```

contains

```

subroutine DzetaSol(Gmax)
implicit real*8(a-h,o-z)
dimension dzeta1(Nx,Ny)
do i=1,Nx
dzeta(i,Ny)=(-3*Uw+4*u(i,Ny-1)-u(i,Ny-2))/dy1
dzeta(i,1)=(-4*u(i,2)+u(i,3))/dy1
end do
do j=1,Ny
dzeta(1,j)=(4*v(2,j)-v(3,j))/dx1
dzeta(Nx,j)=(-4*v(Nx-1,j)+v(Nx-2,j))/dx1
end do
dzeta1=dzeta; Gmax=0.0
do i=2,Nx-1
do j=2,Ny-1
G=-u(i,j)*(dzeta(i+1,j)-dzeta(i-1,j))/dx1 &
-v(i,j)*(dzeta(i,j+1)-dzeta(i,j-1))/dy1 &
+anu*((dzeta(i+1,j)-2*dzeta(i,j)+dzeta(i-1,j))/dx2 &
+(dzeta(i,j+1)-2*dzeta(i,j)+dzeta(i,j-1))/dy2)
if(Gmax<abs(G))Gmax=abs(G)
dzeta(i,j)=dzeta1(i,j)+dt*G
end do
end do
end subroutine DzetaSol

```

```

subroutine PsiSol(it)
implicit real*8(a-h,o-z)
dimension psi1(Nx,Ny)
do k=1,itmax
psi1=psi
do i=2,Nx-1
do j=2,Ny-1

```

```

f=-dzeta(i,j)
psi(i,j)=si1*(psi1(i+1,j)+psi(i-1,j)) &
      +si2*(psi1(i,j+1)+psi(i,j-1)) &
      -si3*f
psi(i,j)=(1-omega)*psi1(i,j)+omega*psi(i,j)
end do
end do
dpsi=maxval(abs(psi-psi1))
if(dpsi<eps) exit
end do
it=k
end subroutine PsiSol

```

```

subroutine UVCalc()
implicit real*8(a-h,o-z)
do i=2,Nx-1
do j=2,Ny-1
u(i,j)=...
v(i,j)=...
end do
end do
end subroutine UVCalc

```

```

subroutine DIVCalc(divmax)
implicit real*8(a-h,o-z)
divmax=-1.
do i=2,Nx-1; do j=2,Ny-1
div=...
if(div>divmax)divmax=div
end do; end do
end subroutine DIVCalc

```

```

end

```

```

logical function SOLID(x,y)
implicit real*8(a-h,o-z)
SOLID=.false.
!if((x>=0.5).and.(y>=0.3))SOLID=.true.
end

```

```

subroutine Vectors1(Nx,Ny,x,y,Vx,Vy,c,icol,ivec,dr,ivar)
use Graflib
implicit real*8(a-h,o-z)
parameter (angle=0.3,tip=0.40)
dimension x(Nx),y(Ny),Vx(Nx,Ny),Vy(Nx,Ny)
logical SOLID
Vmax = sqrt(maxval(Vx**2+Vy**2)); np=50
do i=2,Nx-1,ivec
do j=2,Ny-1,ivec

```



```

if(SOLID(x(i),y(j))) then
  cycle
else
  if(ivar==1)then
    Vmod=sqrt(Vx(i,j)**2+Vy(i,j)**2)
    dx = c*Vx(i,j)/Vmod
    dy = c*Vy(i,j)/Vmod
  else
    dx = c*Vx(i,j)/Vmax
    dy = c*Vy(i,j)/Vmax
  end if
! dx=c*Vx(i,j)/Vmax; dy=c*Vy(i,j)/Vmax ! -old
  x2=x(i)+dx; y2=y(j)+dy
  call Plot2((/x(i),x2/),(/y(j),y2/),icol,np,dr)
  cs=cos(angle); sn=sin(angle)
  dxi=-dx; dyi=-dy
  x3=x2+( dxi*cs+dyi*sn)*tip
  y3=y2+(-dxi*sn+dyi*cs)*tip
  x4=x2+( dxi*cs-dyi*sn)*tip
  y4=y2+( dxi*sn+dyi*cs)*tip
  call Plot2((/x2,x3/),(/y2,y3/),icol,np,dr)
  call Plot2((/x2,x4/),(/y2,y4/),icol,np,dr)
end if
end do
end do
end

```

