

```
// example 10 : RayleighBenard.edp [slide page 55]
// Rayleigh-Benard thermal convection in a box
// P2/P1/P2 element with Characteristic Galerkin
// time evolution data will be stored in "rb.data" for Rayleigh-Benard-stat.edp
// for tutorial by Japan SIAM, Tokyo, 11-12 Feb.2016, Atsushi Suzuki
```

```
int n1 = 80;
int n2 = 20;
real Pr = 0.71;
real Ra = 1500.0;
real dt = 0.01;
real alpha = 1.0/(dt * Pr);
int timestepmax = 600;

mesh Th=square(n1,n2,[x*4.0,y]);

fespace Xh(Th,[P2,P2,P1]);
fespace Vh(Th,P2);
fespace Qh(Th,P1);

Xh [u1,u2,p], [v1, v2, q];
Vh up1, up2, th, thp, psi;
Qh pp, ss, rr;
macro d11(u1)      dx(u1) //
macro d22(u2)      dy(u2) //
macro d12(u1,u2)   (dy(u1) + dx(u2))/2.0 //
macro div(u1,u2)   (dx(u1) + dy(u2)) //

real epsln = 1.0e-6; // penalization parameter to avoid pressure ambiguity
int i;
problem NS([u1,u2,p],[v1,v2,q],solver=UMFPACK,init=i) =
  int2d(Th)(alpha * (u1*v1 + u2*v2)
    + 2.0* (d11(u1)*d11(v1)+2.0*d12(u1,u2)*d12(v1,v2)+d22(u2)*d22(v2))
    - p * div(v1, v2) - q * div(u1, u2)
    - p * q * epsln)
- int2d(Th)( alpha * (convect([up1,up2],-dt,up1)*v1
  +convect([up1,up2],-dt,up2)*v2) )
- int2d(Th) (Ra * thp * v2)
+ on(1,3,u2=0)
+ on(2,4,u1=0);

problem Heat(th,psi,solver=UMFPACK,init=i) =
  int2d(Th)(alpha * (th * psi)
    + dx(th) * dx(psi) + dy(th) * dy(psi))
- int2d(Th)(alpha * convect([up1, up2], -dt, thp) * psi)
+ on(1,th=1)
+ on(3,th=0);

problem streamlines(ss,rr,solver=UMFPACK) =
  int2d(Th)( dx(ss)*dx(rr) + dy(ss)*dy(rr))
+ int2d(Th)( rr*(dy(u1)-dx(u2)))
+ on(1,2,3,4,ss=0.0);

plot(Th,wait=1);

u1[] = 0.0; // impulsive start
th = (1.0-y); // conductive solution
for (i = 0; i < timestepmax; i++) {
  up1 = u1;
  up2 = u2;
  pp = p;
  thp = th;
  NS;
  Heat;
  plot(th,value=true);
  streamlines;
  plot(ss,nbiso=30);
  if (i % 20 == 0) {
    plot ([u1,u2],value=true,wait=0,coef=0.1);
  }
}
```

```
}  
// write time-evolution data for RayleighBenrad-stat.edp  
up1=u1;  
up2=u2;  
pp=p;  
thp=th;  
{  
  ofstream file("rb.data", binary);  
  file.precision(16);  
  for (int i = 0; i < up1[.n; i++) {  
    file << up1[i] << " ";  
  }  
  file << endl;  
  for (int i = 0; i < up2[.n; i++) {  
    file << up2[i] << " ";  
  }  
  file << endl;  
  for (int i = 0; i < pp[.n; i++) {  
    file << pp[i] << " ";  
  }  
  file << endl;  
  for (int i = 0; i < thp[.n; i++) {  
    file << thp[i] << " ";  
  }  
  file << endl;  
}
```