Solution to Orr-Sommerfeld Equation

Plane Poiseuille flow of a Newtonian fluid. The eigenvalue is c = c_r + ic_i in this code.
B.C.s : v(-1) = 0, v(1) = 0, v'(-1) =0, v'(1) = 0

In[52]:=
```
Clear["`*"]
k = 1;
Rey = 10000.0;
nG = 201;
za = -1.;
zb = 1.;
```

Creating the Differentiation matrices (d1 and d2)

In[58]:=
```
cA = (zb - za) / 2; cB = (zb + za) / 2; zptoz = (z - cB) / cA;
leszp = -Table[Cos[(π j) / nG], {j, 0, nG}];
lesz = N[cA leszp + cB];
leszInt = Take[lesz, {2, nG}]; di = (2 nG^2 + 1) / 6; lesInt = {2, nG};
cbar = Table[Switch[j, 0, 2, nG, 2, _, 1], {j, 0, nG}];
diag =
  Table[Switch[i, 0, -di, nG, di, _, Cos[(π i) / nG] / (2 (Sin[(π i) / nG]) ^2)], {i, 0, nG}];
d1 = (cA^-1) Table[Switch[i - j, 0, diag[[i + 1]], _, (cbar[[i + 1]] (-1) ^ (i + j + 1)) /
      (cbar[[j + 1]] (Cos[(π i) / nG] - Cos[(π j) / nG]))], {i, 0, nG}, {j, 0, nG}] // N;
d2 = d1.d1;
(*equivalently d2=d1.d1*)
d4 = (d2.d2);
```

In[67]:=
```
basevel = Table[1 - lesz[[i + 1]]^2, {i, 0, nG}]; (* base velocity on each grid point*)
```

In[68]:=
```
matA  = I * k * DiagonalMatrix[basevel].d2 - I * k^3 DiagonalMatrix[basevel] +
   2 * I * k * IdentityMatrix[nG + 1] - 1/ (Rey) (d4 - 2 k^2 d2 + k^4 IdentityMatrix[nG + 1]);
matB  =  I * (d2 - k^2 IdentityMatrix[nG + 1]);
```

In[70]:=
```
Table[matA[[1, j]] = 0, {j, 2, nG + 1}]; matA[[1, 1]] = 1;
Table [matA[[nG + 1, j]]  = 0,  {j, 1, nG}];
matA[[nG + 1, nG + 1]]  = 1;
matA[[2]]  = d1[[1]];
matA[[nG]] = d1[[nG + 1]];

Table[matB[[1, j]] = 0, {j, 1, nG + 1}];
Table [matB[[nG + 1, j]]  = 0,  {j, 1, nG + 1}];
Table[matB[[2, j]]  = 0,  {j,  1,  nG + 1}];
Table[matB[[nG, j]]  = 0,  {j,  1,  nG + 1}];
```

In[81]:=
```
res = Reverse[Eigenvalues[{matA, matB}]];
ListPlot[res /. Complex[a_, b_] → {a, b},
 Frame → True, PlotRange → {{-0.1, 1.1}, {-0.5, 0.02}},
 FrameLabel → {Style["Re(λ)", 18], Style["Im(λ)", 18]},
 PlotLabel → "Most dangerous mode:\n λ=" <>
   ToString[NumberForm[res[[1]], NumberFormat → (Row[{#1, "e", #3}] &)]]]
```

Out[82]=