Numerical Solver of 2D Infinite Well

This script is a numerical Solver for the 2D infinite well scenario utilising finite differences of the second order derivative.

Preparing Values

m is mass, where here it is set to that of the electron rest mass in $eV \cdot s^2 \cdot m^{-2}$.
A is the width of the infinite well in m^2.
a is the width of the segments of the infinite well for the finite differences approximation, in the same units.

```
m = 0.511 * 10^6 / (9 * 10^16);
A = 0.53 * 10^-10;
```

n is the number of steps segmenting the domain of the infinite well.

```
n = 6;
```

```
a = A / (n + 1);
```

node is the *n*th harmonic to be displayed by the graphing solution below.

```
node = 5;
```

Preparing Eigensystem

```
ij = Flatten[Table[{i, j}, {j, n}, {i, n}], 1];
```

```
getMatrixElement[x_, y_, n_] :=
  If[x == y, 4,
   If[x == y - 1, -1,
    If[x == y + 1, -1,
     If[x == y - n, -1,
      If[x == y + n, -1, 0]]]]];
```

Here we construct the matrix resulting from applying the finite differences approximation to the TISE.

```
finiteDifferencesMatrix = Table[
   Table [
    getMatrixElement[j, i, n]
    , {j, n^2}]
   , {i, n^2}];
```

Solving Eigenproblem and Plotting Results

Solve the eigenproblem and obtain the resulting eigensystem.

```
eigensystem = Eigensystem[finiteDifferencesMatrix];
```

This shows the first harmonic of the wave function for a particle in the constructed infinite well, and its respective energy.

```
plot = Table[Join[ij[[k]], {eigensystem[[2, -node, k]]}], {k, n^2}];
For[x = 0, x < (n + 2), x++, AppendTo[plot, {x, 0, 0}]];
For[x = 0, x < (n + 2), x++, AppendTo[plot, {x, n + 1, 0}]];
For[x = 1, x < (n + 1), x++, AppendTo[plot, {0, x, 0}]];
For[x = 1, x < (n + 1), x++, AppendTo[plot, {n + 1, x, 0}]];
```

Plot results!

```
ListPlot3D[plot]
```