PROGRAMMING ASSIGNMENT 4

Problem 1. You are asked to write a Laplacian solver based on solving a system of linear equations as in HW 6 problem 3. The idea is to build a sparse matrix M for the system $Mx = b$, as well as the inhomogeneous vector b. The key to writing such a program is to first work out by hand a few small examples. Use the `laplace_solver_rdm_walk.ipynb` notebook as a guide. You can find it in Canvas under Coding Files. To get you started, here are the first few lines of the main loop:

```
# build matrix M and the right-hand side b


b = np.zeros( (n**2, 1) )

I = n*[0]
J = n*[0]
V = n*[0]

for i in range(n):

    for j in range(n):

        # index of u(i,j) in the arrays
        k = i + j*n

        # Check if the node's lower neighbor is a boundary node.
        if i > 0: # no
            I.append(k)
            J.append(k-1)
            V.append(0.25)


        else:     # yes
            b[k] = b[k] - g_bottom/4.0
```