ECSE597/ECSE472
Assignment #2

In order to simplify parsing, we will use our own netlist format in ECSE597/ECSE472. For example consider the following netlist.

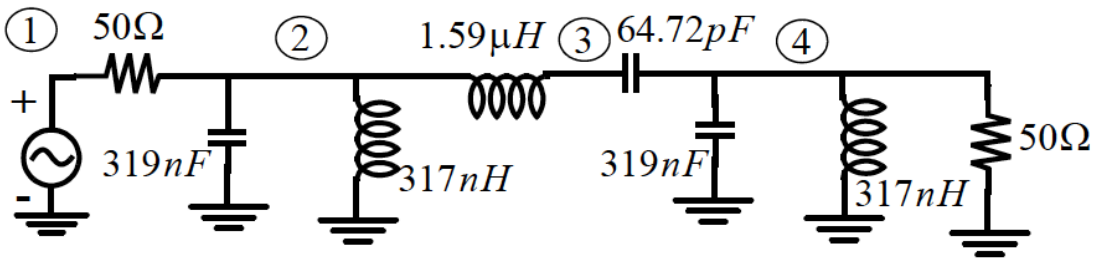
```
% CIR1.M
% Description of Circuit 1
% Author: R. K.
% Date: Sept 17, 2019
% Comment lines start with % just like in matlab (this is in fact a matlab
% script).

global G C %define global variables
global b;

G = zeros(4,4); % Define G, 4 node circuit (do not include additional
variables)
C = zeros(4,4); % Define C, 4 node circuit (do not include additional
variables)
b = zeros(4,1); % Define b, 4 node circuit (do not include additional
variables)
% Try changing zeros(4,4) with sparse(4,4) for sparse routines.
% See matlab help for info about sparse matrix support.


vol(1,0,1); % add voltage source between nodes 0 and 1 (value =1)
res(1,2,50); % add 50 ohm resistor between nodes 1 and 2
res(4,0,50);
cap(2,0,0.319e-6);
cap(3,4,64.72e-12);
cap(4,0,0.319e-6);
ind(2,0,0.317e-6);
ind(2,3,1.59e-6);
ind(4,0,0.317e-6);
```

The above netlist corresponds to the following circuit which has 4 nodes.



You will find on myCourses the following functions: cap.m, res.m, cur.m, vol.m, and diode.m which allow you to generate the MNA equations for circuits containing capacitors, resistors, current sources, voltage sources and diodes. You will also find the function f_vector.m which computes the nonlinear vector of the MNA, f(x), as a function of x. The file Circuit_diodeckt1.m is the netlist of a simple diode circuit. nlJacobian.m and dcsolve.m are the definitions of functions that you will need to write. Finally, Test_bench_diode1.m is the test bench that you should be able to run once you complete the assignment.

Questions:

1. Draw the circuit in the netlist Circuit_diodeckt1.m.
2. Write a matlab function nlJacobian.m (see function definition provided). This function should compute and return the Jacobian of the nonlinear vector f(x)
3. Write a matlab function dcsolve.m (see function definition provided). This function should use the Newton-Raphson method to compute the dc solution. It should also return the interim values of detlaX for each iteration (see function definition for details).
4. Run the script Test_bench_diode1.m to test your code.

Deliverables:

1. A pdf file containing a) the schematic of the circuit in Circuit_diodeckt1.m, b) dc values computed after running test bench, c) the figure that was plotted when you ran the test bench, and d) the matlab code for nlJacobian.m and dcsolve.m.
2. The matlab m files for nlJacobian.m and dcsolve.m