

Assignment 9, PHYS 490
Computers in Physics
Due 4/8/10 at start of class

In class, you have re-explored the “relaxation” technique mentioned by Dr. Trantham back in the C unit with MATLAB. This last homework problem will explore this further. One of the issues associated with running Dr. Trantham’s “relax” code was that there were some issues with the edges. There are a number of ways that this can be addressed; one of them is with so-called “periodic boundary conditions”. When you need to look further to the left for an averaging technique and there *is* no element to the left, you take the rightmost element on the same row. In short, you use copies of the whole domain to handle the boundaries. (Dr. Larsen should explain this pictorially in class). This problem is meant to explore this notion of periodic boundary conditions and examine your technique on creating a relaxation code similar to that developed in class.

You need to generate a function (called relax) that takes in as an argument a matrix. You may assume the matrix should be 200 x 200. Most of this matrix will be zeros, but there will be some elements that make up electrodes like those done in HW 6 (e.g. twolines or octagon or something similar). You need to leave the electrodes alone, but do a relaxation on the rest of the array (with periodic boundary conditions), to generate the steady-state solution to Poisson’s equation. Other inputs are (1) the number of relaxation steps, and (2) a variable that – if equal to true (i.e. 1) will create a movie of each step in the process and if equal to false (i.e. 0) will not create a movie.

Outputs will be the relaxed matrix, the largest change of any matrix element in the last relaxation step, and – if requested – a variable containing the movie.

You have more time to work on this assignment, and it is just a single problem, so I am going to try and “break” your code. Make sure you have a complete and descriptive help file. Make sure you have some sort of conditions if the user tries putting in an invalid input. Make sure you comment excessively. Make sure you’ve tried to account for anything that a user (malicious or not) may try to do.