**Math 3626**: Applied Matrix Theory

**Linear Programming HW**

In class we discussed the Simplex algorithm to solve linear optimization problems. We also created functions in Mathematica that apply the simplex algorithm to the matrix known as the “initial tableau” and solve it. In particular, we created the functions getPivotCol, getPivotRow, and doPivot to implement various steps in the Simplex algorithm. Download the two Mathematica worksheets on linear programming / simplex method and review/test the functions to make sure they work correctly. Note that after defining these functions we created a code using a While loop to repeat the various steps of the algorithm to find the solution.

1. Create a function “doSimplex” that takes as input a matrix A representing an initial tableau, sets the variable “tableau” to the input matrix A, and then calls our existing functions in a while loop until the algorithm is over; it should then display the final tableau from which you can read off the answer to the Simplex method.
2. Use your function doSimplex to verify the solution of the following optimization problems:
   1. Maximize 4 x + 2 y subject to 10 x + 5 y <= 50, 12 x + y <= 48, 3 x + 2 y <= 18, x >= 0, and y >= 0.  
      Answer: Max = 20, x -> 19/5, y -> 12/5
   2. Maximize 2 x - 3 y + 4 z subject to 4 x - 3 y + z <= 3, x + y + z <= 10, 2 x + y - z <= 10, x >= 0, y >= 0, and z >= 0  
      Answer: Max = 111/4, x -> 0, y -> 7/4, z -> 33/4

1. Try your code on this problem:   
   *Maximize 4 x + 2 y subject to 10 x + 5 y <= 50, y <= 48, 3 x + 2 y <= 18, x >= 0, and y >= 0*   
   It will fail (unless you made some modification to the above functions). Explain what goes wrong and exactly which of your functions fails and exactly why and at what step. Then discuss at least one possibility to fix the problem, in words. Finally, when you are convinced that your fix works, implement it and try your new algorithm again on the above example. If your fix works, your program should now deliver the final tableau from which you can read off the solution (the solution actually is the same as for the similar problem before.

**Extra Credit**: For *extra credit*, create a function that displays the answer to the optimization problem in some easy-to-read output. Use that function at the end of doSimplex to produce the answer in a nice, readable way. **Note**: You get half of the extra credit points for describing your solution in words, and the other half for creating the actual code to add to your doSimplex function.

For *additional extra credit*, modify the doSimplex program so that you don’t have to enter the entire initial tableau but only the coefficient matrix A and a vector representing the constants on the right.