GLY 5826

Assignment 3

1. Beginning with the 1-D Laplace equation ($\frac{\partial^2 h}{\partial x^2} = 0$), write a general analytical expression for h(x). Using the boundary conditions $h(0) = 0$ m and $h(100) = 1$ m, solve for the constants in your general expression and write the specific solution.

2. Repeat 1 but change the BCs to $dh/dx|_{x=0} = 10^{-3}$ and $h(100) = 1$.

3. Derive the finite difference expression for the 1-D Laplace equation ($\frac{\partial^2 h}{\partial x^2} = 0$). Show all steps.

4. In 1 you used the boundary conditions $h(0) = 0$ m and $h(100) = 1$ m, to determine the analytical solution to the 1-D Laplace equation. Now solve the same problem using a spreadsheet finite difference model. Plot the results from both approaches on the same graph; use open symbols for the numerical solution and a solid line for the analytical solution.

5. Repeat 4 but change the BCs to $dh/dx|_{x=0} = 10^{-3}$ and $h(100) = 1$.

6. Create a MATLAB ‘m-file’ of the lines below. Add contour labels. Add a quiver plot showing the direction of ground water flow. Note that you need to work with the transpose of h (i.e., rows and columns switched; h’ in Matlab) to compute the gradient components.

```
% Wang and Anderson Equation 2.1 (Toth (1962) regional solution)


    % Clear memory and close any open figures

    clear
    close('all')

    % set up x and y loops; print i so you know progress

    for i=1:101
       i
      x=i-1
       for j = 1:51
         y=j-1
```

```
        % Refer to Figures 1.6 and 1.7
        % c is total head gradient along top
        c=.001;

        % s is length of domain
        s=100;

        % y0 is elevation at x = 0
        y0=50;

        % Do summation
        % zero accumulator term
        summation=0;
        % loop through 1000 terms
        for m=0:100

summation=summation+cos((2*m+1)*pi*x/s)*cosh((2*m+1)*pi*y/s)/((2*m+1)^2*co
sh((2*m+1)*pi*y0/s));

        end

        % Compute h(i,j)

        h(i,j)=(y0+c*s/2)-4*c*s/(pi^2)*summation;

    end

 end

% Plot
contour([0:100],[0:50],h',[50.005:0.005:50.1])
axis('equal')
```