% Mini Project: Fluid Mechanics(CHO-102): Solving Navier-Stokes Equation with Suitable Assumptions
% Compiled by: **Ojasvi Tripathi** (22045181) on Matlab, using some resources and pre-existing functions on Matlab

```
% Define parameters
Lx = 1.0;         % Length of the domain in the x-direction
Ly = 1.0;         % Length of the domain in the y-direction
Nx = 51;          % Number of nodes in the x-direction
Ny = 51;          % Number of nodes in the y-direction
dx = Lx / (Nx - 1); % Grid spacing in the x-direction
dy = Ly / (Ny - 1); % Grid spacing in the y-direction
f = zeros(Nx, Ny);  % Source term (zero for Navier-Stokes equation)
tol = 1e-6;         % Tolerance for convergence
maxiter = 1000;     % Maximum number of iterations

% Define boundary conditions
phi_left = 0;             % Potential on the left boundary
phi_right = 0;            % Potential on the right boundary
phi_bottom = 0;           % Potential on the bottom boundary
phi_top = sin(pi * (0:Ny-1) * dy); % Potential on the top boundary

% Initialize potential function
phi = zeros(Nx, Ny);
phi(1, :) = phi_left;
phi(Nx, :) = phi_right;
phi(:, 1) = phi_bottom;
phi(:, Ny) = phi_top;

% Solve Poisson equation using Gauss-Seidel iteration
iter = 0;        % Iteration counter
err = inf;       % Error norm

while err > tol && iter < maxiter
    iter = iter + 1;
    phi_old = phi; % Store old values of potential

    % Loop over interior nodes
    for i = 2:Nx - 1
        for j = 2:Ny - 1
            % Update potential using five-point stencil
            dphi_dx2 = (phi(i + 1, j) - 2 * phi(i, j) + phi(i - 1, j)) / (dx^2);
            dphi_dy2 = (phi(i, j + 1) - 2 * phi(i, j) + phi(i, j - 1)) / (dy^2);
            phi(i, j) = 0.5 * (dphi_dx2 + dphi_dy2) - dx * dy * f(i, j);
        end
    end

    % Compute error norm
    err = max(max(abs(phi - phi_old)));
end

% Display results
disp(['Number of iterations: ', num2str(iter)])
disp(['Error norm: ', num2str(err)])

% Plot potential function
[X, Y] = meshgrid((0:Nx - 1) * dx, (0:Ny - 1) * dy); % Create grid for plotting
figure(1)
surf(X, Y, phi)
```

```
xlabel('x')
ylabel('y')
zlabel('\phi')
title('Potential function')
```