# Assignment 06

**Name:** Manav Kothari

**UIN:** 133008008

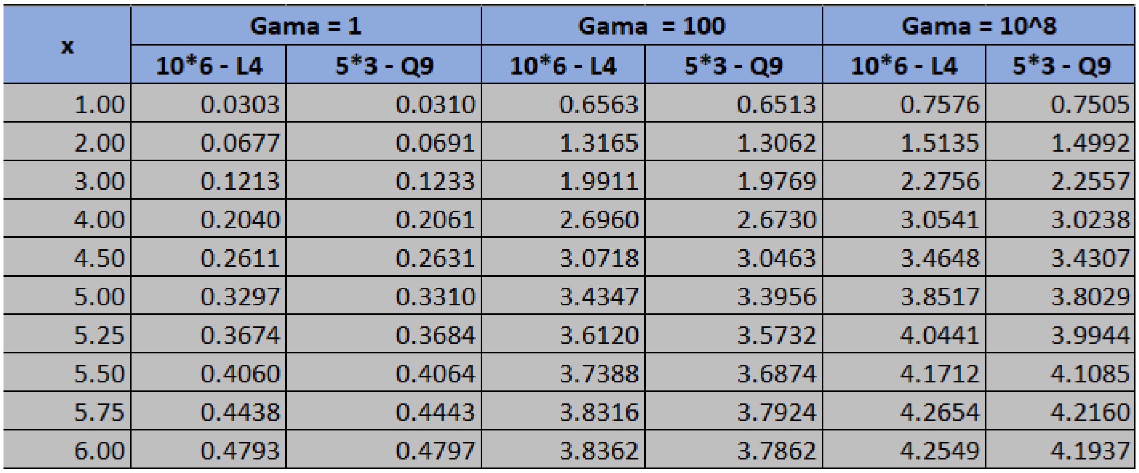
## Problem 10.8.1

### Inputs

| Parameter | Value |
|---|---|
| Simulation Input | |
| Problem | 1 |
| NONLIN | 0 (Linear) |
| ITERMAX | 25 |
| Epsilon | 0.001 |
| GAMA1 (Acceleration Parameter) | 0.5 |
| GAMA2 (Penalty Term) | 10^8 |
| Domain Input | |
| X0 | 0 |
| Y0 | 0 |
| X_length | 6 |
| Y_length | 2 |
| Mesh Input | |
| NX | 10 |
| NY | 6 |
| NPE | 4 |
| DX | [1 1 1 1 0.5 0.5 0.25 0.25 0.25 0.25] |
| DY | [0.25 0.25 0.5 0.5 0.25 0.25] |
| NDF | 2 |
| IEL | 1 |
| NGPF | 2 |
| NGPR | 1 |
| Loading Condition | |
| DP | 1 |
| F = [FX FY] | [0 0] |
| MU | 1 |
| Essential Boundary Conditions | |
| NSPV | 39 |
| ISPV | [1 77 76 75 74 73 72 71 70 69 68 67 56 45 34 23 12 1 2 34 5 6 7 8 9 10 11 77 76 75 74 73 72 71 70 69 68 67; 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2]' |
| VSPV | [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1]' |
| | |

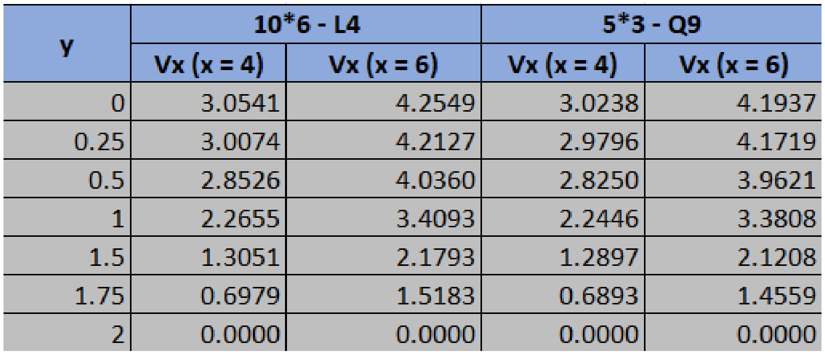
| Natural Boundary Condition | |
|---|---|
| NSSV | 31 |
| ISSV | [1 2 3 4 5 6 7 8 9 10 11 22 33 44 55 66 77 1 11 22 33 44 55 66 77 67 56 45 34 23 12; 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2]' |
| VSSV | zeros(max(size(ISSV)),1) |

## Results



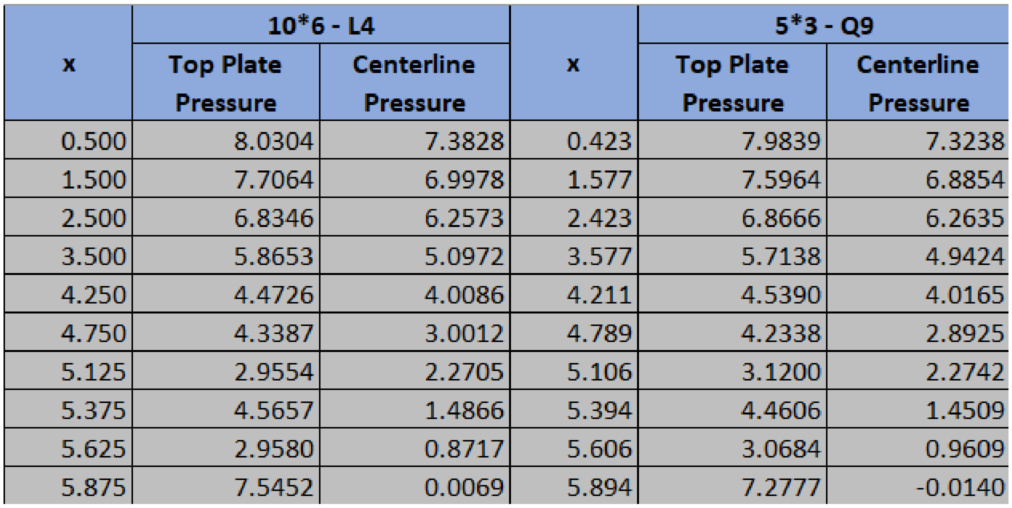
| x | Gama = 1 | | Gama = 100 | | Gama = 10^8 | |
|---|---|---|---|---|---|---|
| | 10*6 - L4 | 5*3 - Q9 | 10*6 - L4 | 5*3 - Q9 | 10*6 - L4 | 5*3 - Q9 |
| 1.00 | 0.0303 | 0.0310 | 0.6563 | 0.6513 | 0.7576 | 0.7505 |
| 2.00 | 0.0677 | 0.0691 | 1.3165 | 1.3062 | 1.5135 | 1.4992 |
| 3.00 | 0.1213 | 0.1233 | 1.9911 | 1.9769 | 2.2756 | 2.2557 |
| 4.00 | 0.2040 | 0.2061 | 2.6960 | 2.6730 | 3.0541 | 3.0238 |
| 4.50 | 0.2611 | 0.2631 | 3.0718 | 3.0463 | 3.4648 | 3.4307 |
| 5.00 | 0.3297 | 0.3310 | 3.4347 | 3.3956 | 3.8517 | 3.8029 |
| 5.25 | 0.3674 | 0.3684 | 3.6120 | 3.5732 | 4.0441 | 3.9944 |
| 5.50 | 0.4060 | 0.4064 | 3.7388 | 3.6874 | 4.1712 | 4.1085 |
| 5.75 | 0.4438 | 0.4443 | 3.8316 | 3.7924 | 4.2654 | 4.2160 |
| 6.00 | 0.4793 | 0.4797 | 3.8362 | 3.7862 | 4.2549 | 4.1937 |

Table 1: Vx (x,0)



| y | 10*6 - L4 | | 5*3 - Q9 | |
|---|---|---|---|---|
| | Vx (x = 4) | Vx (x = 6) | Vx (x = 4) | Vx (x = 6) |
| 0 | 3.0541 | 4.2549 | 3.0238 | 4.1937 |
| 0.25 | 3.0074 | 4.2127 | 2.9796 | 4.1719 |
| 0.5 | 2.8526 | 4.0360 | 2.8250 | 3.9621 |
| 1 | 2.2655 | 3.4093 | 2.2446 | 3.3808 |
| 1.5 | 1.3051 | 2.1793 | 1.2897 | 2.1208 |
| 1.75 | 0.6979 | 1.5183 | 0.6893 | 1.4559 |
| 2 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |

Table 2: Horizontal velocity Vx (x,y) vs vertical distance y



| x | 10*6 - L4 | | x | 5*3 - Q9 | |
|---|---|---|---|---|---|
| | Top Plate Pressure | Centerline Pressure | | Top Plate Pressure | Centerline Pressure |
| 0.500 | 8.0304 | 7.3828 | 0.423 | 7.9839 | 7.3238 |
| 1.500 | 7.7064 | 6.9978 | 1.577 | 7.5964 | 6.8854 |
| 2.500 | 6.8346 | 6.2573 | 2.423 | 6.8666 | 6.2635 |
| 3.500 | 5.8653 | 5.0972 | 3.577 | 5.7138 | 4.9424 |
| 4.250 | 4.4726 | 4.0086 | 4.211 | 4.5390 | 4.0165 |
| 4.750 | 4.3387 | 3.0012 | 4.789 | 4.2338 | 2.8925 |
| 5.125 | 2.9554 | 2.2705 | 5.106 | 3.1200 | 2.2742 |
| 5.375 | 4.5657 | 1.4866 | 5.394 | 4.4606 | 1.4509 |
| 5.625 | 2.9580 | 0.8717 | 5.606 | 3.0684 | 0.9609 |
| 5.875 | 7.5452 | 0.0069 | 5.894 | 7.2777 | -0.0140 |

Table 3: Pressure vs Horizontal distance x (Near top plate & centerline)

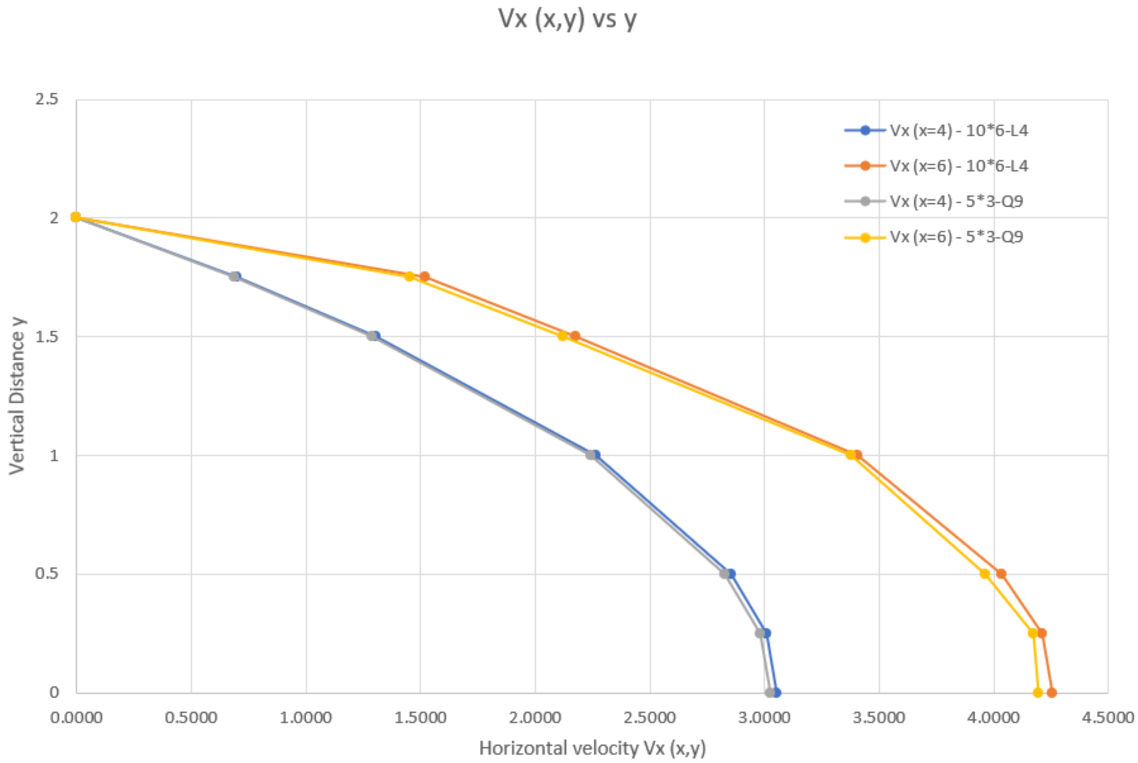


Figure 1: Horizontal Velocity field Vx at x = 4 & x = 6

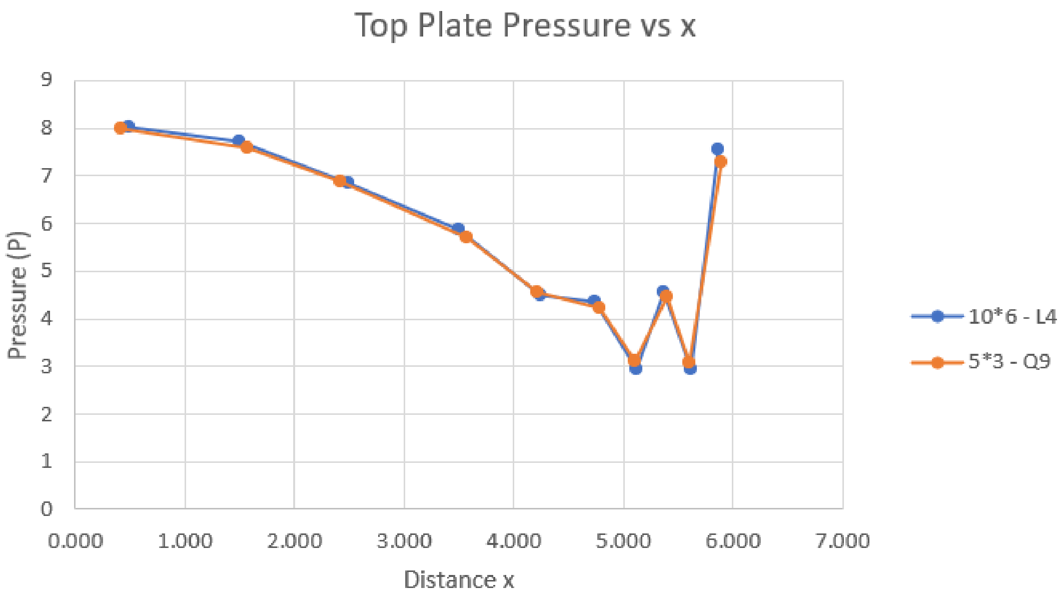


Figure 2: Top Plate Pressure vs x

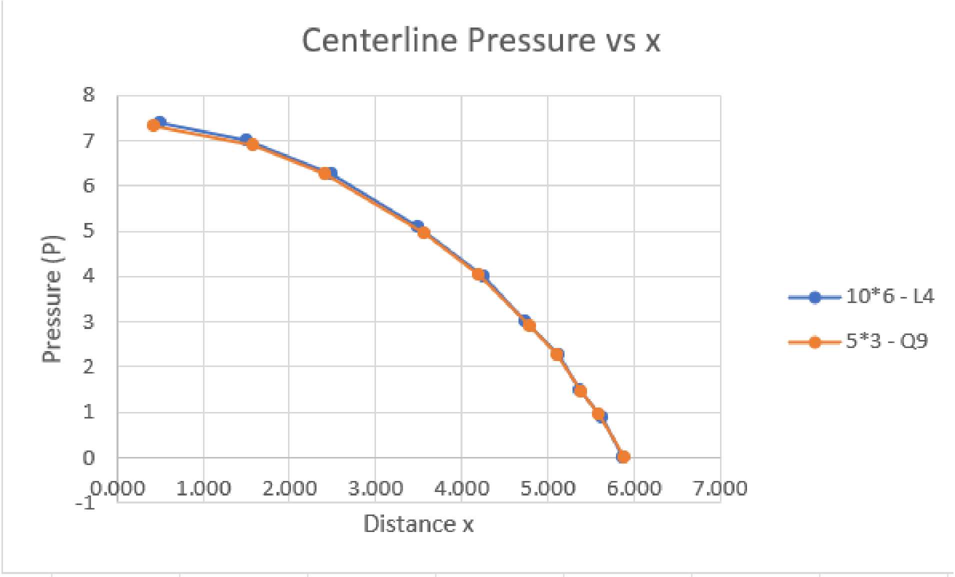


Figure 3: Centerline Pressure vs x

# Problem 10.8.4

## Inputs

| Parameter | Value |
|---|---|
| Simulation Input | |
| Problem | 2 |
| NONLIN | 1 |
| ITERMAX | 25 |
| Epsilon | 0.001 |

| | |
|---|---|
| GAMA1 (Acceleration Parameter) | 0.5 |
| GAMA2 (Penalty Term) | 10^8 |
| Domain Input | |
| X0 | 0 |
| Y0 | 0 |
| X_length | |
| Y_length | 1 |
| Mesh Input | |
| NX | 8 |
| NY | 10 |
| NPE | 9 |
| DX | 0.125*ones(1,NX) |
| DY | [0.125 0.125 0.125 0.125 0.125 0.125 0.0625 0.0625 0.0625 0.0625] |
| NDF | 2 |
| IEL | 2 |
| NGPF | 3 |
| NGPR | 2 |
| Loading Condition | |
| DP | [0 500 1000] |
| F = [FX FY] | [0 0] |
| MU | 1 |
| Essential Boundary Conditions | |
| NSPV | 144 |
| ISPV | [1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 34 51 68 85 102 119 136 153 170 187 204 221 238 255 272 289 306 323 340 357 356 355 354 353 352 351 350 349 348 347 346 345 344 343 342 341 324 307 290 273 256 239 222 205 188 171 154 137 120 103 86 69 52 35 18 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 34 51 68 85 102 119 136 153 170 187 204 221 238 255 272 289 306 323 340 357 356 355 354 353 352 351 350 349 348 347 346 345 344 343 342 341 324 307 290 273 256 239 222 205 188 171 154 137 120 103 86 69 52 35 18; 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2]' |
| VSPV | [0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]' |

| Natural Boundary Condition | |
|---|---|
| NSSV | 0 |
| ISSV | [ ] |
| VSSV | [ ] |

## Results

| y | Mesh: 8*8 - L4 | | | | | | | Mesh: 4*4 - Q9 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Re -> | Linear | 250 | | 500 | | 750 | | Linear | 250 | | 500 | | 750 | |
| | | DI | NI | DI | NI | DI | NI | | DI | NI | DI | NI | DI | NI |
| 0.125 | -0.0579 | -0.0367 | -0.0367 | -0.0239 | -0.0235 | -0.0128 | -0.0121 | -0.0615 | -0.0412 | -0.0410 | -0.0131 | -0.0120 | 0.0146 | 0.0151 |
| 0.250 | -0.0988 | -0.0688 | -0.0689 | -0.0502 | -0.0498 | -0.0320 | -0.0310 | -0.1039 | -0.0851 | -0.0848 | -0.0520 | -0.0502 | 0.0017 | 0.0031 |
| 0.375 | -0.1317 | -0.0944 | -0.0947 | -0.0733 | -0.0732 | -0.0533 | -0.0526 | -0.1393 | -0.1283 | -0.1283 | -0.1133 | -0.1119 | -0.0481 | -0.0459 |
| 0.500 | -0.1471 | -0.0911 | -0.0915 | -0.0696 | -0.0701 | -0.0569 | -0.0573 | -0.1563 | -0.1305 | -0.1311 | -0.1284 | -0.1295 | -0.1086 | -0.1079 |
| 0.625 | -0.0950 | -0.0176 | -0.0177 | 0.0043 | 0.0037 | 0.0020 | 0.0010 | -0.1118 | -0.0437 | -0.0442 | -0.0494 | -0.0517 | -0.0901 | -0.0908 |
| 0.750 | 0.0805 | 0.0469 | 0.0479 | 0.0414 | 0.0414 | 0.0323 | 0.0322 | 0.0481 | 0.0753 | 0.0753 | 0.1042 | 0.1042 | 0.0549 | 0.0517 |
| 0.875 | 0.4500 | 0.2616 | 0.2617 | 0.1712 | 0.1714 | 0.1207 | 0.1198 | 0.4186 | 0.2833 | 0.2838 | 0.2139 | 0.2133 | 0.1495 | 0.1482 |

Table 4: velocity Vx(0.5,y) for different Reynold’s number

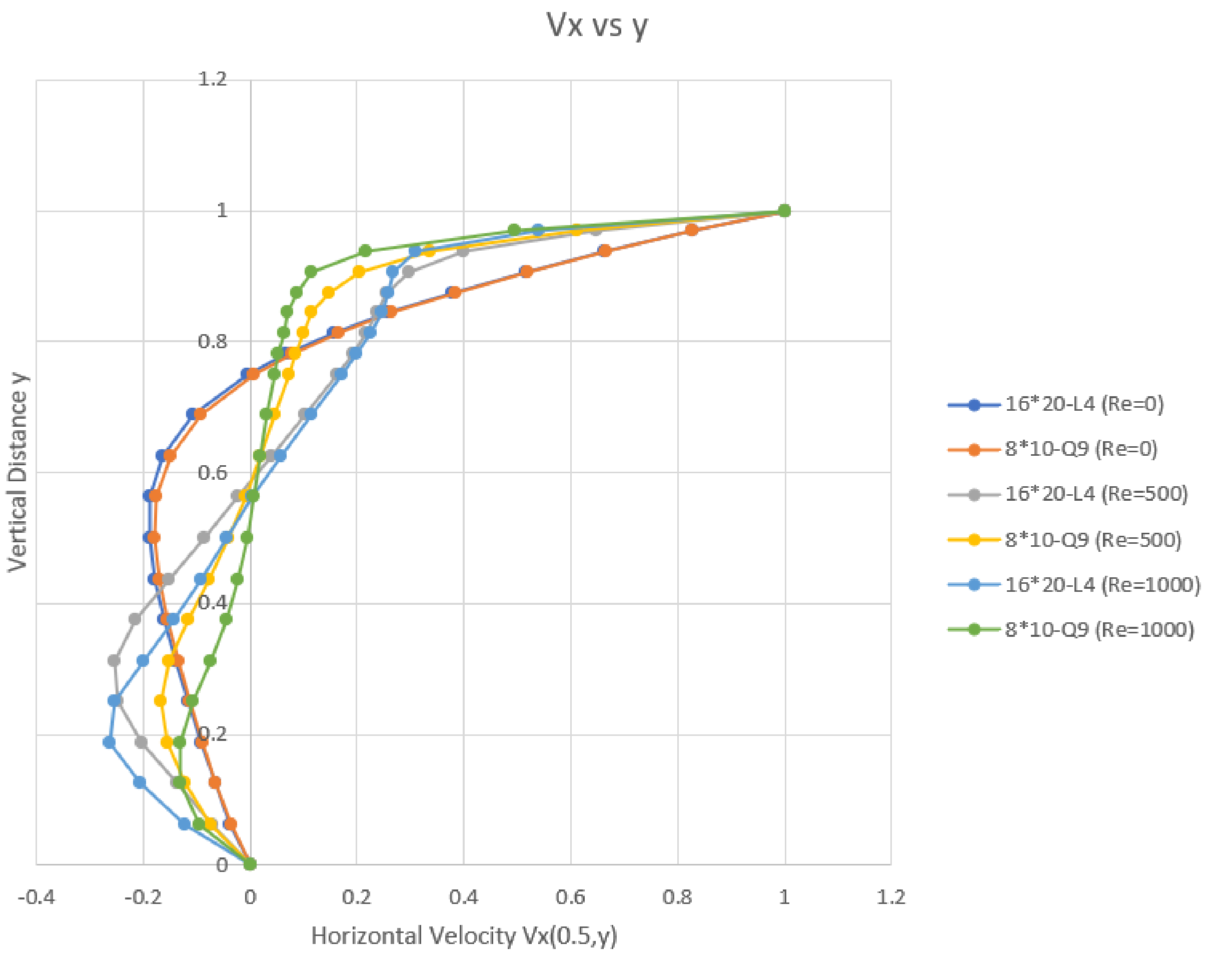


Figure 4: Horizontal Velocity Vx(0.5,y) vs y for Re = 0,500,1000

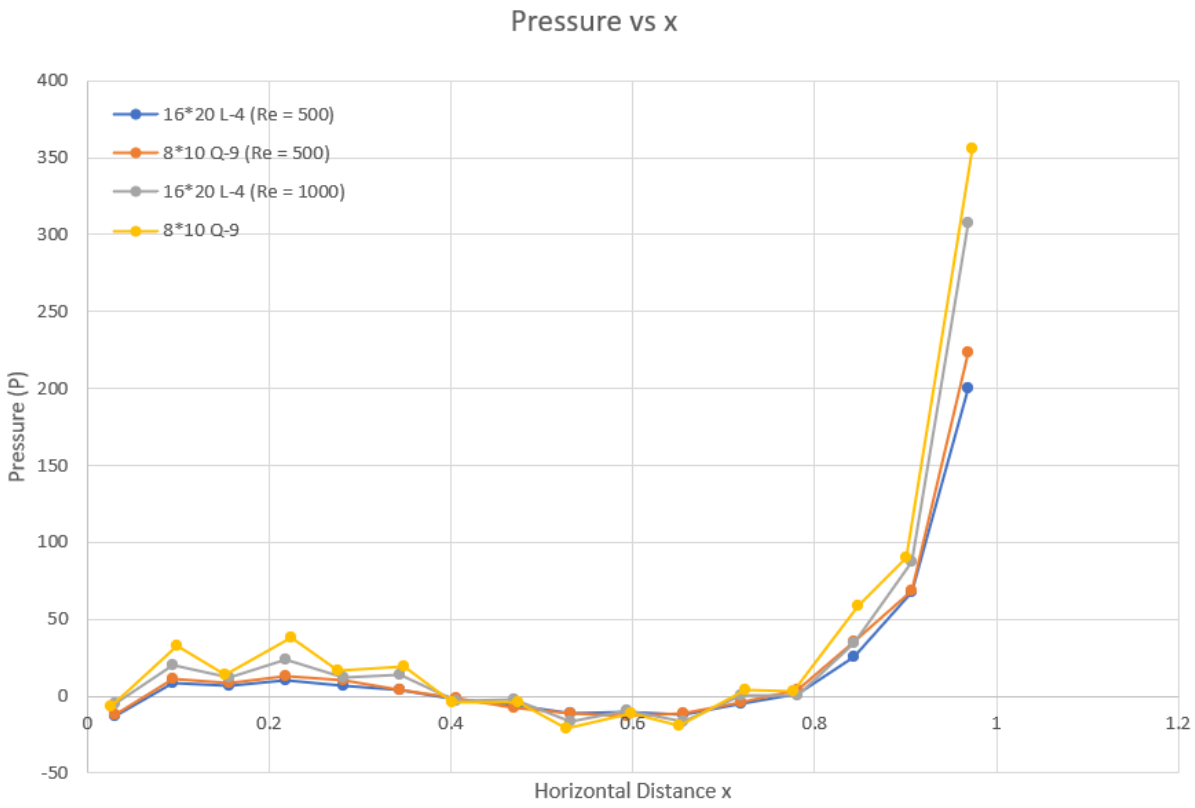


Figure 5: Plot of Pressure P along the top wall of the cavity

```
%% Pseudo Code
% Do not run this

Give all Inputs

Call MeshR function to generate 2D Rectangular Mesh
Call PlotMesh function to get visual reprsentation of the↙
generated mesh

Define DOF_NOD matrix
% DOF_NOD matrix is but a matrix similar to NOD matrix, but↙
rather than
% mapping global node number with element number, it maps↙
global degree of
% freedom with element number

Initialize current and previous iteration solution (GCU & GPU)

%% Big Loop

for NL = 1:NLS % Initiate load step loop
    Define required constants like Rho and Mu
    while iter<=ITERMAX && convergence == 0 % Initiate↙
iterative loop
        Initialize GLK and GLF matrices
        for N = 1:NEM
            Calculate ELU matrix from GPU & GCU & Acceleration↙
parameter (GAMA1)

            Define ELXY from GLXY

            Call FLUIDMATRICS function to calculate ELK and ELF

            Perform Assembly of ELK & ELF into GLK & GLF
        end
        Call FLUIDBCS function to apply Essential and Natural↙
BCs

        Calculate current iteration solution GCU

        Null out VSPV for NI after 1st iteration

        Calculate error and check for convergence

        iter = iter+1;
    end % END OF ITERATIVE LOOP

    %% Post Processing of converged solution

    if IGRAD ~= 0 % Check if post-processing is required by the↙
user or not
        for I = 1:NEM % Calculate for all elements
            Define ELXY and ELU

            Call STRESS2D function to calculate Pressure for↙
all the gaussian points in the element

            Call Press_Calc script to print the required↙
Pressure values
            % Note Press_Calc is question specific and needs to↙
be changed
            % if we solve any other question (except 10.8.1 &↙
10.8.4)

        end
    end
end % END OF LOAD STEP LOOP

PRINT SOLUTIONS
```

```
%↙
---------------------------------------------------------------↙
------- %

function [ELK,ELF] = FLUIDMATRICS(NDF,NPE,NONLIN,ELXY,ELU,↙
RHOAMU,NGPF,GAMA2)

Initialize ELK and ELF

% Full integration
for NI = 1:NGPF
    for NJ = 1:NGPF
        Calculate ELF and part of ELK matrix

        if NONLIN > 1
            Calculate TANG matrix as well
        end
    end
end

% Reduced Integration
for NI = 1:NGPR
    for NJ = 1:NGPR
        Calcuate the penalty term in ELK
    end
end

if NONLIN > 1
    Calculate final TANG matrix and Residual matrix
end
end

%↙
---------------------------------------------------------------↙
```

```
---------%

function [XMAT,YMAT,SXMAT,SYMAT,SXYMAT,PRSMAT] = STRESS2D(ELXY,↙
NPE,ELU,NGPR,GAMA2,MU)

% Perform Reduced integration
for NI = 1:NGPR
    for NJ = 1:NGPR
        Find the requried Gauss points where Pressure needs to↙
be calculated in an element
        Calculate corresponding pressure.
    end
end
```