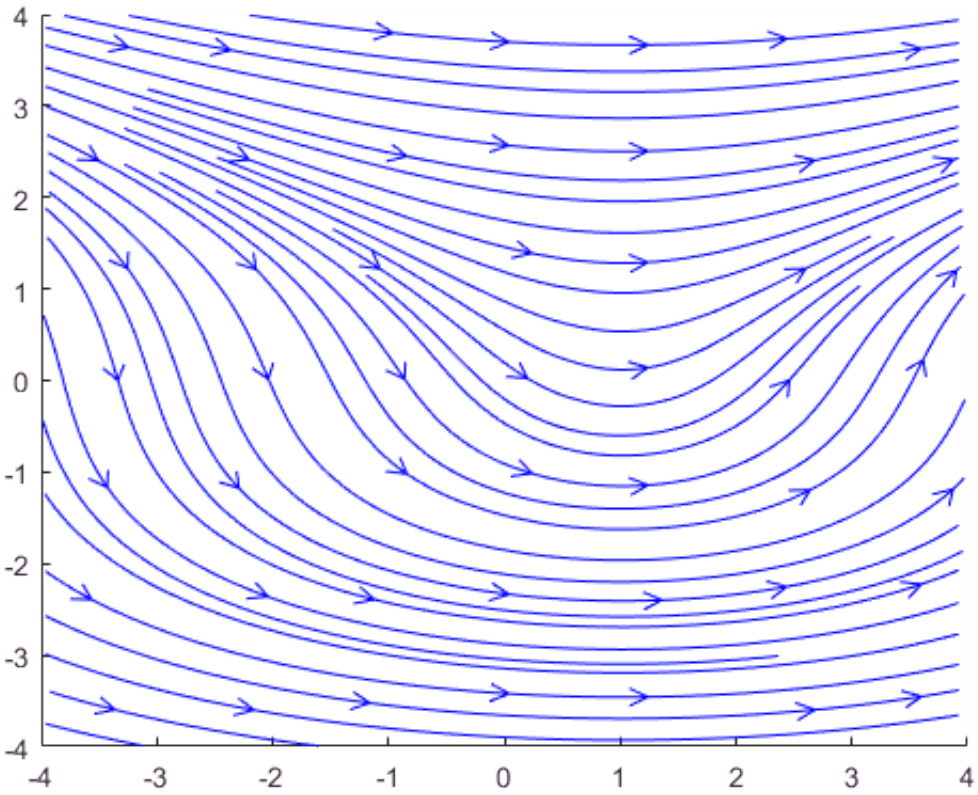
Question - 1

```
clc;
clear;
close all;

syms x y c;

u(x, y) = 1 + y^2;
v(x, y) = x - 1;

stream_line = int(u, y) == int(v, x) + c;
plot_range = -4:0.4:4;
[x, y] = meshgrid(plot_range, plot_range);

streamslice(x, y, u(x, y), v(x, y))
```



Question - 2

```
clc;
clear;

syms x y;

u(x, y) = (x^2 / 2) - (x^3 / 3);
v(x, y) = x * (x - 1) * (y + 1);
```

```
motion_cond = diff(u, y) - diff(v, x);
rotational_cond = diff(u, x) + diff(v, y);

if motion_cond == 0
   disp('Motion/Flow is possible')
else
   disp('Motion/Flow is not possible')
end
```

```
Motion/Flow is not possible
```

```
if rotational_cond == 0
   disp('Motion/Flow is rotational')
else
   disp('Motion/Flow is not irrotational')
end
```

```
Motion/Flow is not irrotational
```

```
[x, y] = solve(u == 0, v == 0, x, y);
stagnation_points = [x, y];
stagnation_points
```

```
stagnation_points =
```

$$\begin{pmatrix} \frac{3}{2} & -1 \\ 0 & 0 \end{pmatrix}$$

Question - 3

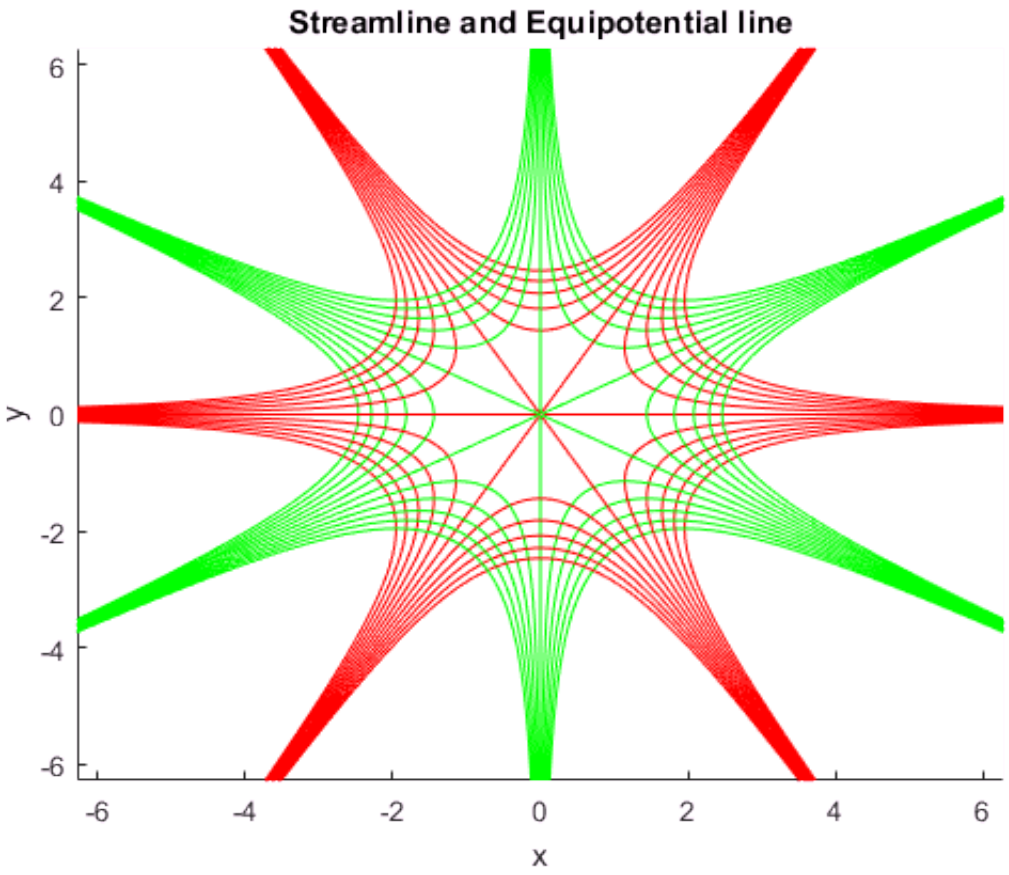
```
clc;
clear;
close all;

syms x y;

a = 1;
u = a * (x^2 - y^2);
v = - 2 * a * x * y;

stream_exists = diff(u, x) + diff(v, y);
velocity_potential_exists = diff(u, y) - diff(v, x);

if stream_exists == 0
   disp('Stream Function exists')
else
   disp('Stream Function does not exists')
end
```

```
Stream Function exists
```

```
if velocity_potential_exists == 0
   disp('Velocity Potential exists')
else
   disp('Velocity Potential does not exists')
end
```

```
Velocity Potential exists
```

```
psi = int(u, y);
psi = psi + int(diff(psi, x) + v, x);

phi = int(u, x);
phi = phi + int(diff(phi, y) - v, y);

for i = -5:5
    hold on

    f1 = ezplot(psi == i);
    set(f1, 'color', 'r')

    f2 = ezplot(phi == i);
    set(f2, 'color', 'g')

    title('Streamline and Equipotential line')
end
```



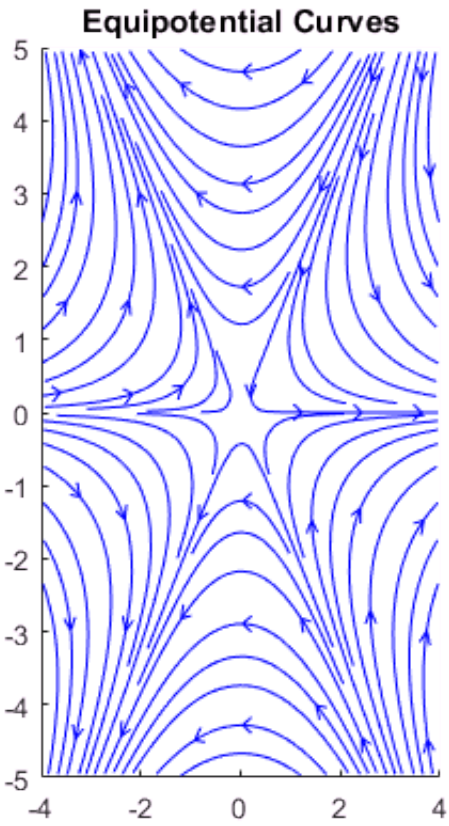
Question - 4

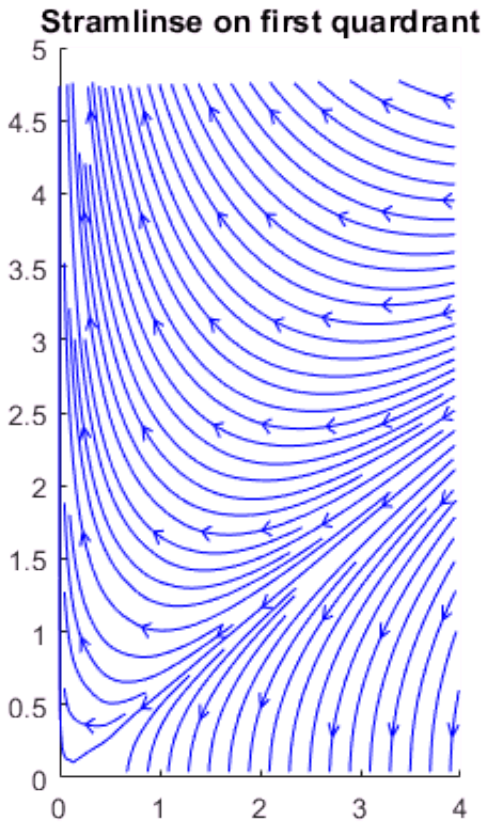
```
clc;
clear;
close all;

syms x y;

a = 2;
phi(x, y) = (a * x^3) / 3 - (a * x * y^2) - 2;

u = diff(phi, x);
v = diff(phi, y);

% a
subplot(1, 2, 1)
[x, y] = meshgrid(-4:0.4:4, -5:0.4:5);

streamslice(x, y, u(x, y), v(x, y))
title('Equipotential Curves')
% b
syms x y c;
stream_func = int(u, y) - int(v, x) + c;
stream_func
```

```
stream_func(x, y) =
```

$$4\,x^2\,y - \frac{2\,y^3}{3} + c$$

```
%c
subplot(1, 2, 2);

[x, y] = meshgrid(0:0.4:4, 0:0.4:5);
streamslice(x, y, v(x, y), -u(x, y))
title('Stramlinse on first quardrant')
```





Question - 5

```
clc;
clear;

a = 0.6;
Cc = 0.61;
z1 = 5;
g = 9.81;
z2 = Cc .* a;

az2 = a / z1;
if 0.1 < az2 && az2 < 0.2

    p = z2 .* sqrt((2 .* g .* (z1 - z2))/(1 - (z2/z1).^2));
    fprintf('Flowrate is: %f\n', p)
else
    disp('Try changing value of a')
end
```

```
Flowrate is: 3.499252
```

```
z1 = 5:0.25:15;
q = zeros();

for i = 1:length(z1)
    q(i) = z2 .* sqrt((2 .* g .* (z1(i) - z2))/(1 - (z2/z1(i)).^2));
```

```
end

table(z1', q', 'VariableNames', {'z1', 'Qb'})
```

```
ans =
     z1        Qb
    _____    ______

        5    3.4993
     5.25    3.5915
      5.5    3.6815
     5.75    3.7693
        6    3.8552
     6.25    3.9392
      6.5    4.0215
     6.75    4.1022
        7    4.1813
     7.25     4.259
      7.5    4.3353
     7.75    4.4102
        8     4.484
     8.25    4.5565
      8.5    4.6279
     8.75    4.6983
        9    4.7676
     9.25    4.8359
      9.5    4.9032
     9.75    4.9697
       10    5.0353
    10.25       5.1
     10.5     5.164
    10.75    5.2271
       11    5.2896
    11.25    5.3512
     11.5    5.4122
    11.75    5.4725
       12    5.5322
    12.25    5.5912
     12.5    5.6496
    12.75    5.7074
       13    5.7647
    13.25    5.8213
     13.5    5.8774
    13.75     5.933
       14    5.9881
    14.25    6.0427
     14.5    6.0968
    14.75    6.1504
       15    6.2036
```

Question - 6

```
clc;
clear;
close all;
syms x y c r t;

m = -0.314;

m2pi = m /2 * pi;
```

```
u1 = diff(m2pi .* log(sqrt(x^2 + (y - 5)^2)), x);
v1 = diff(m2pi .* log(sqrt(x^2 + (y - 5)^2)), y);

u2 = diff(m2pi .* log(sqrt(x^2 + (y + 5)^2)), x);
v2 = diff(m2pi .* log(sqrt(x^2 + (y + 5)^2)), y);

u = simplify(u1 + u2);
v = simplify(v1 + v2);

stream_func = int(u, y) - int(v, x) + c;
stream_func_polar = subs(stream_func, {x, y}, {r * cos(t), r * sin(t)});

[x, y] = meshgrid([-6:0.5:-0.5 0.5:0.5:6], [-6:0.5:-0.5 0.5:0.5:6]);

stream_func
```

```
stream_func =
```

$$c + \frac{157\,\pi\,\operatorname{atan}\left(\frac{x}{y-5}\right)}{1000} + \frac{157\,\pi\,\operatorname{atan}\left(\frac{x}{y+5}\right)}{1000} - \frac{157\,\pi\,\left(\operatorname{atan}\left(\frac{y}{x}-\frac{5}{x}\right)+\operatorname{atan}\left(\frac{y}{x}+\frac{5}{x}\right)\right)}{1000}$$

```
stream_func_polar
```

```
stream_func_polar =
```

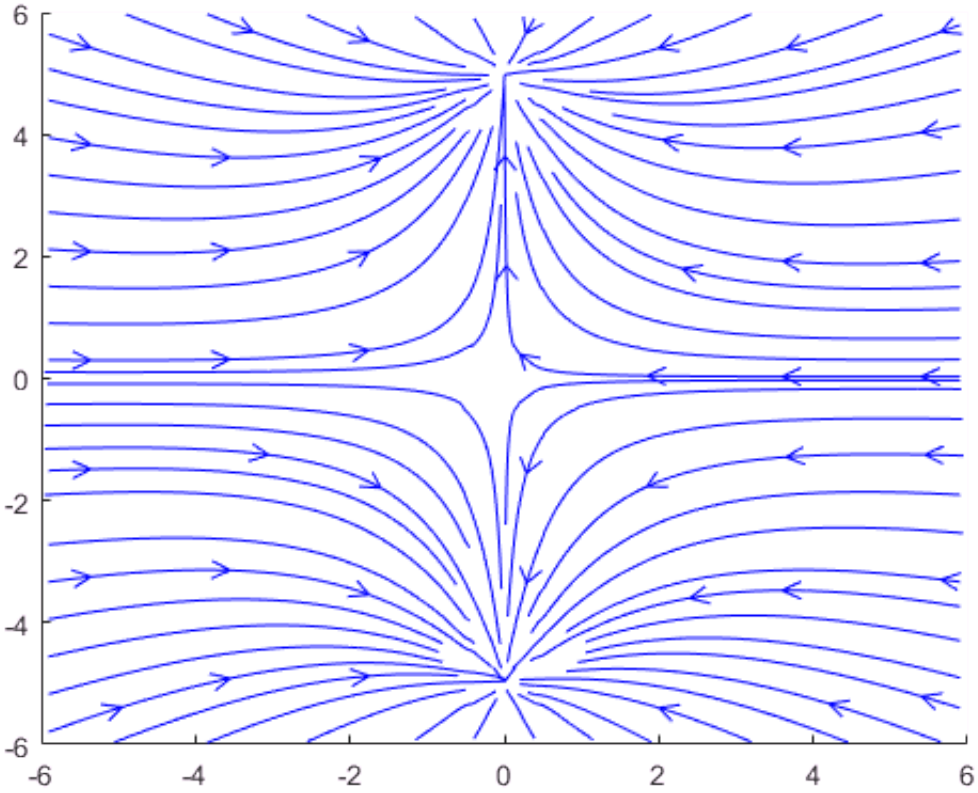
$$c + \frac{157\,\pi\,\operatorname{atan}\left(\frac{r\cos(t)}{r\sin(t)-5}\right)}{1000} + \frac{157\,\pi\,\operatorname{atan}\left(\frac{r\cos(t)}{r\sin(t)+5}\right)}{1000} - \frac{157\,\pi\,\left(\operatorname{atan}(\sigma_2+\sigma_1)-\operatorname{atan}(\sigma_2\right.}{1000}$$

where

$$\sigma_1 = \frac{\sin(t)}{\cos(t)}$$

$$\sigma_2 = \frac{5}{r\cos(t)}$$

```
streamslice(x, y, subs(u), subs(v))
```



Question - 7

```
clc;
clear;
syms x;

Mersenne(x) = 2^x - 1;
Fermat(x) = 2^vpa(2).^x + 1;

n = (1:10)';
Mn = double(Mersenne(n));
Fn = double(Fermat(n));
Prime_Mn = isprime(Mn);
Prime_Fn = isprime(Fn);

table(n, Mn, Prime_Mn, Fn, Prime_Fn)
```

```
ans =
    n      Mn     Prime_Mn        Fn         Prime_Fn
    __    ____    ________    __________    ________

     1       1    false                5    true
     2       3    true                17    true
     3       7    true                65    false
     4      15    false              257    true
     5      31    true              1025    false
     6      63    false             4097    false
     7     127    true             16385    false
```

```
     8     255    false            65537    true
     9     511    false         2.6215e+05    false
    10    1023    false         1.0486e+06    false
```

Question - 8

```
clc;
clear;

n = 1250;

f = factor(n);
p = unique(f);
s = size(p, 2);

a = zeros();
for i = 1:s
   a(i) = nnz(f(1, :) == p(1, i));
end

tau = 1;
sigma = 1;
phi = n;

for i = 1:s

    tau = tau * (a(1, i) + 1);
    sigma = sigma * (p(1, i)^(a(1, i) + 1) - 1) / (p(1, i) - 1);
    phi = phi * (1 - (1 / p(1, i)));
end

table(tau, sigma, phi)
```

```
ans =
    tau    sigma    phi
    ___    _____    ___

    10     2343     500
```

Question - 9

```
clc;
clear;

n = 8;
R = zeros();

if mod(n, 2) == 0

    for k = 1:n-1
        for i = 1:n

            m = mod(k - i, n - 1);
```

```
            if k == n - 1 && i == n - 1
               R(k, i) = n;
            elseif k == n - 1 && i == n
                R(k, i) = n - 1;
            elseif m == i
                R(k, i) = n;
                C = i;
            elseif i == n
                R(k, i) = C;
            elseif m == 0
                R(k, i) = n - 1;
            else
                R(k, i) = m;
            end
        end
    end
else
    for k = i:n
        for i = 1:n

            m = mod(k - i, n);

            if k == n - 1 && i == n - 1
               R(k, i) = n;
            elseif k == n && i == n
                R(k, i) = 0;
            elseif m == i
                R(k, i) = 0;
                C = i;
            elseif m == 0
                R(k, i) = n;
            else
                R(k, i) = m;
            end
        end
    end
end

col = {strings()};
row = {strings()};

col_name = 'Team%d';
row_name = 'Round-%d';

for i = 1:n
   col(i) = {sprintf(col_name, i)};
   row(i) = {sprintf(row_name, i)};
end

array2table(R, 'RowNames', row(1:size(R, 1)), 'VariableNames', col)
```

```
ans =
               Team1    Team2    Team3    Team4    Team5    Team6    Team7    Team8
               _____    _____    _____    _____    _____    _____    _____    _____

    Round-1    7        6        5        8        3        2        1        4
    Round-2    8        7        6        5        4        3        2        1
    Round-3    2        1        7        6        8        4        3        5
    Round-4    3        8        1        7        6        5        4        2
    Round-5    4        3        2        1        7        8        5        6
```

Round-6	5	4	8	2	1	7	6	3
Round-7	6	5	4	3	2	1	8	7