

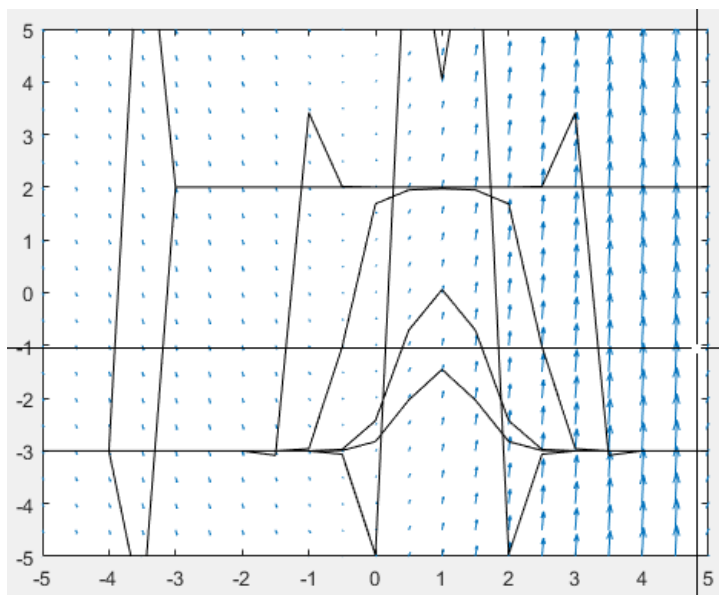
Задача 1:

Код:

```
% Векторно поле, допирателно до интегралните криви на уравнението:
%  $y' = (y - 2) * (y + 3) * (x - 1)$ 
[x, y] = meshgrid(-5:0.5:5);
u = ones(size(x));
e = 2.71828182846;
v = 1 + 10 .* e.^(x./5) .* sin(2 .* x ./ 5);
quiver(x, y, u, v);
axis([-5,5,-5,5]);

% Интегралните криви на уравнението:
%  $y' = (y - 2) * (y + 3) * (x - 1)$ 
hold on;
axis([-5,5,-5,5]);
[a, b] = ginput(1);
syms x y(x)
eq1 = diff(y, x) == (y-2).*(y+3).*(x-1);
while a >= -5 && a <= 5 && b >= -5 && b <= 5
    ic1 = y(a) == b;
    sol1 = dsolve(eq1, ic1);
    xx = -5:0.5:5;
    yy = double(subs(sol1, 'x', xx));
    plot(xx, yy, 'k');
    [a,b] = ginput(1);
end
```

Графика:



## Задача 2:

Код:

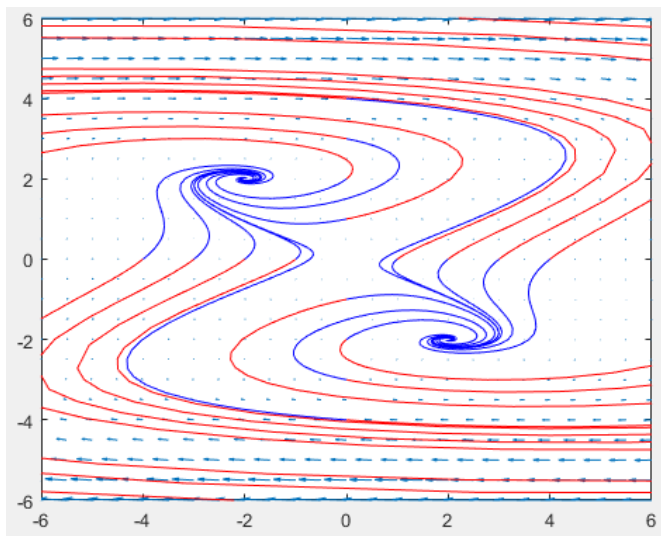
```
[x, y] = meshgrid(-6:0.5:6);
u1 = y.^3 - 5.*y - x;
u2 = -x - y;
quiver(x, y, u1, u2);

% равновесни точки на системата
[t,x] = ode45(@solver,[0,4],[2,0]);
hold on
plot(x(:,1), x(:,2), 'r');
axis([-6,6,-6,6]);
[t,x] = ode45(@solver,[0,-4],[2,0]);
plot(x(:,1), x(:,2), 'g');
axis([-6,6,-6,6]);

% фазов портрет
a = [-4,-3,-2,-1,1,2,3,4,0,0,0,0,0,0,0,0];
b = [0,0,0,0,0,0,0,0,0,-4,-3,-2,-1,1,2,3,4];
hold on
for k = 1:length(a)
    [t,x] = ode45(@solver, [0,4],[a(k), b(k)]);
    plot(x(:,1), x(:,2),'b');
    [t,x] = ode45(@solver, [0,-4],[a(k), b(k)]);
    plot(x(:,1), x(:,2), 'r');
end
axis([-6,6,-6,6]);
```

```
function res = solver(t, x)
res = [x(2)^3 - 5*x(2) - x(1);
      - x(1) - x(2)];
end
```

Графика:



### Задача 3:

#### Код:

```
% Интервал за t
t = 1:0.1:16;

% Интервал за x
x = 0:0.05:8;

% Изобразяване на движението на струна
for k = 1:length(t)
    % Пълно решение на задачата за струната
    u = (phi(x + t(k)) + phi(x - t(k))) / 2 + intpsi(x - t(k), x + t(k)) / 2;
    plot(x, u);

    % Интервал на изобразяване
    axis([0,8,-4,4]);

    % Анимация на диаграмата
    M(k) = getframe;
end

% Повторение на анимацията 3 пъти
movie(M,3);
```

```
% Главната функция, която решава задачата - огледен образ на струната
function res = phi(x)
    for k=1:length(x)
        if x(k) >= 0 && x(k) <= 8
            res(k) = phi_mirror(x(k));
        else if x(k) < 0
            res(k) = -phi(-x(k));
        else
            res(k) = -phi(16-x(k));
        end
    end
end
```

```
% Функция, която дава второстепенната част на решението
function res = intpsi(a,b)
    for k = 1:length(a)
        if a(k) == b(k)
            res(k) = 0;
        else
            x = a(k):(b(k)-a(k))/200:b(k);
            y = phi(x);
            res(k) = trapz(x,y);
        end
    end
end
```

```
% Главната функция, която решава задачата
function res = phi_mirror(x)
    for k = 1:length(x)
        if x(k) >= 3/2 && x(k) <= 7/2
            res(k) = cos(pi * x(k))^2;
        else
            res(k) = 0;
        end
    end
end
```

#### Графика:

