Київський національний університет імені Тарас Шевченка

Факультет комп’ютерних наук та кібернетики

Лабораторна робота №2

з курсу

«Управління динамічними системами»

на тему:

**«Аналітичне конструювання регуляторів.**

**Побудова фазових портретів»**

**Виконав:**

**Студент групи ІПС-21**

**факультету комп’ютерних**

**наук та кібернетики**

**Дубина А.В.**

Київ 2023

Зміст

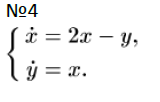
Умови задач згідно варіанту.......................................................................................3

Представлення розв’язку аналітично (в зошиті) .....................................................4

Код програми...............................................................................................................7

Screen з відповідними результатами роботи програми............................................9

**Згідно з варіантом розв’язати наступні приклади:**



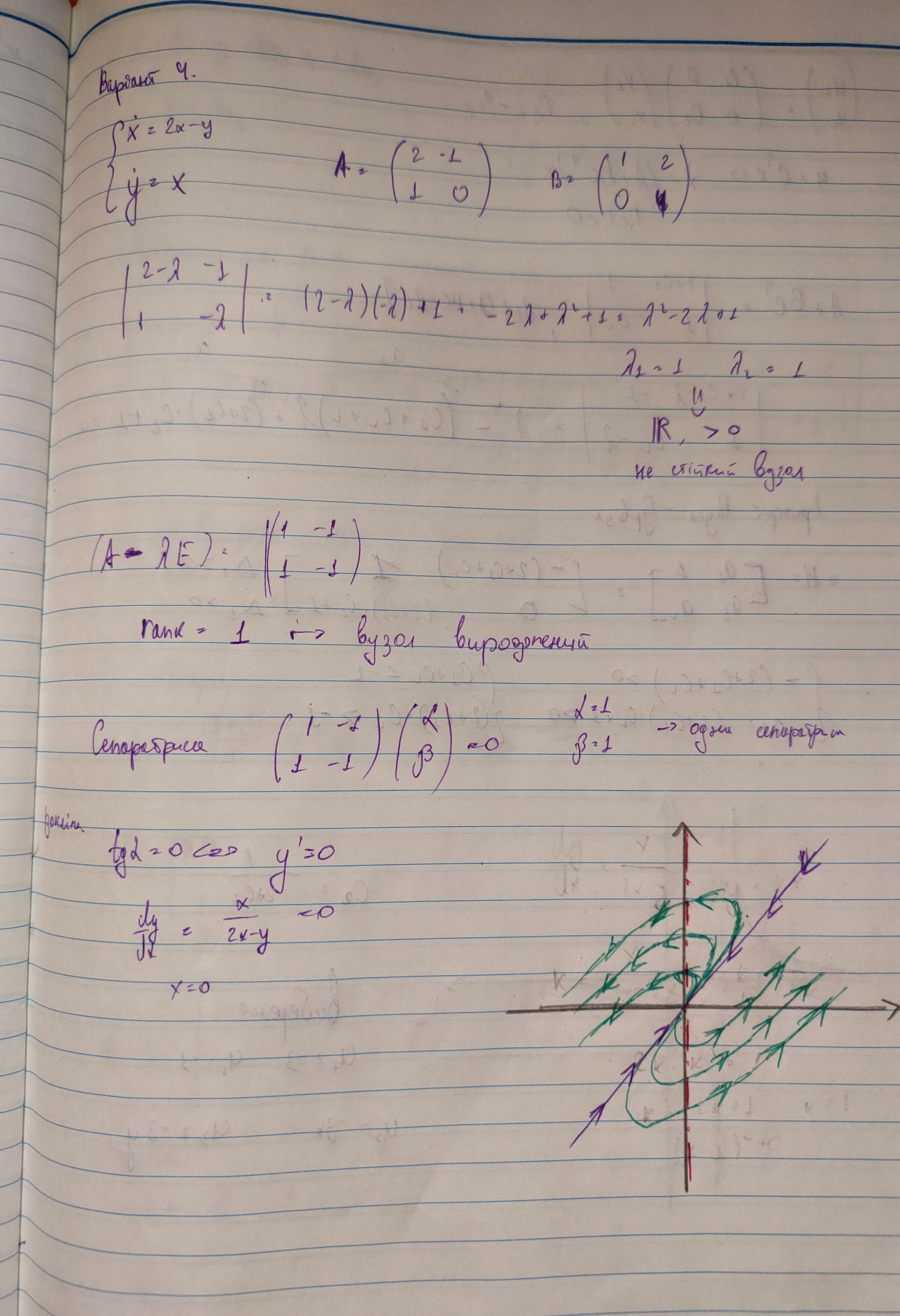
- Дослідити на стійкість задану систему. Визначити вигляд точки спокою. Намалювати фазовий портрет. (Все аналітично в зошиті).

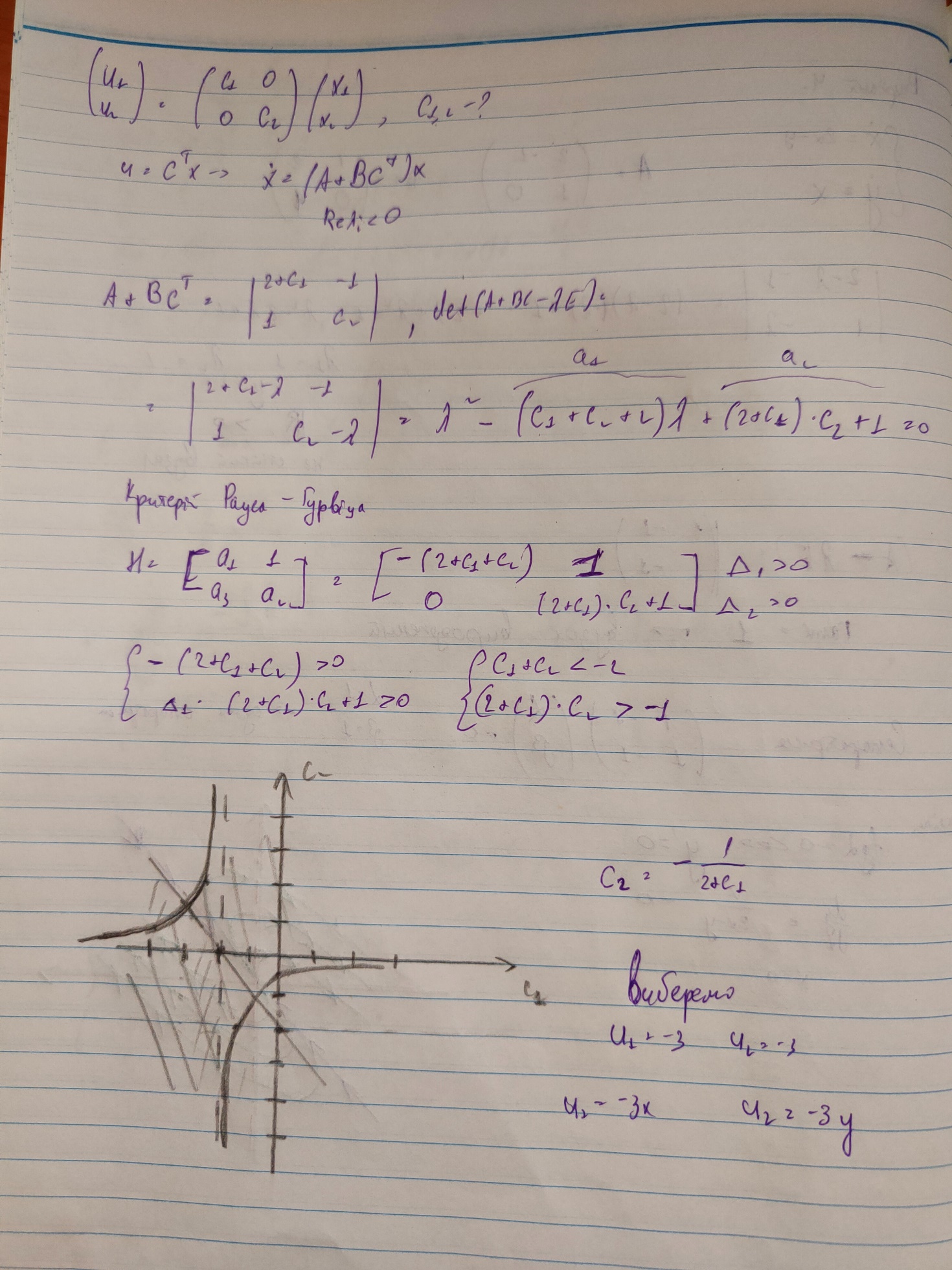
- Розв’язати задачу аналітичного конструювання регуляторів, обравши одне керування з знайдених можливих. Визначити вигляд отриманої точки спокою. Намалювати фазовий портрет. (Все аналітично в зошиті).

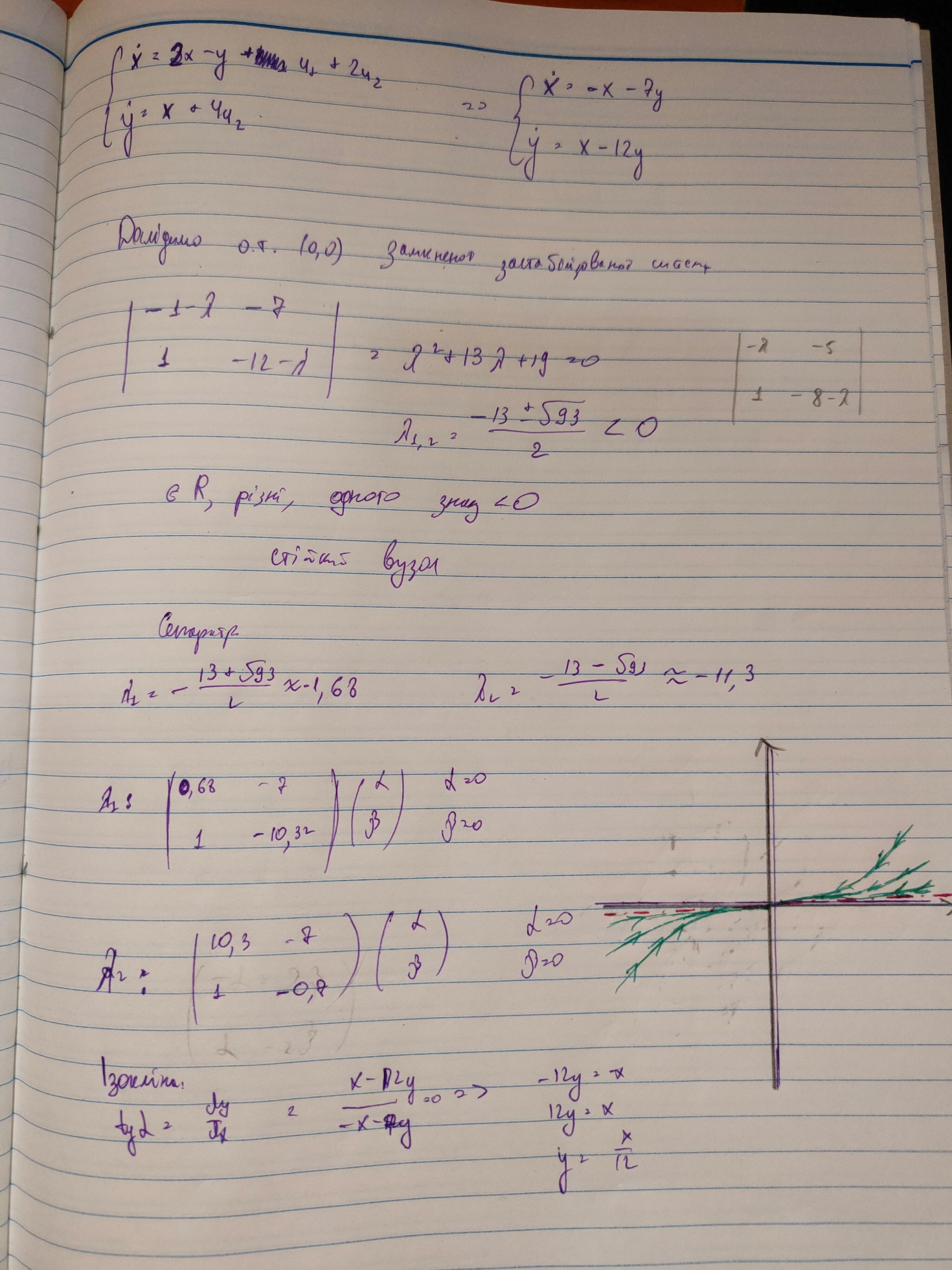
- Зобразити фазові портрети особливих точок розімкненої системи та побудованої замкненої системи за допомогою програмних пакетів (бажано **Sage**). Траєкторії, сепаратриси, ізокліни (де треба) – різний колір та товщина.

Значення матриці B згідно номеру варіанту:

)







Код для розімкненої системи:   
#Записуємо систему

t = var('t')

x1 = function('x')(t)

y1= function('y')(t)

dx = diff(x1, t) == 2\*x1 - y1

dy = diff(y1, t) == x1

#Будуємо розв’язок задачі Коші

x, y = var ('x y')

plot = plot\_vector\_field([2\*x - y, x], [x, -10, 10], [y, -10, 10], axes\_labels=['$x$','$y(x)$'])

for i in range(-10, 10):

x\_t, y\_t = desolve\_system([dx, dy], [x1, y1], ics = [0, -5\*i, -0.5\*i])

plot += parametric\_plot((x\_t.rhs(), y\_t.rhs()), (t, -2\*pi, 2\*pi), color = 'green')

#Будуємо сепаратрису

separ = implicit\_plot(x-y, (x, -10, 10), (y, -10, 10), color = 'blue')

plot += separ

#Будуємо ізокліну

isocline = implicit\_plot(x, (x, -10, 10), (y, -10, 10), linewidth = 1, color = 'red', linestyle = 'dashdot')

plot += isocline

#Виводимо побудований фазовий портрет

plot.show(xmin = -10, xmax = 10, ymin = -10, ymax = 10, axes=true)

Код для замкненої системи:

#Записуємо систему

t = var('t')

x1 = function('x')(t)

y1= function('y')(t)

dx = diff(x1, t) == -x1 - 7\*y1

dy = diff(y1, t) == x1 - 12\*y1

#Будуємо розв’язок задачі Коші

x, y = var ('x y')

plot = plot\_vector\_field([-x - 7\*y, x - 12\*y], [x, -10, 10], [y, -10, 10], axes\_labels=['$x$','$y(x)$'])

for i in range(-10, 10):

x\_t, y\_t = desolve\_system([dx, dy], [x1, y1], ics = [0, -10\*i, -5\*i])

plot += parametric\_plot((x\_t.rhs(), y\_t.rhs()), (t, -2\*pi, 2\*pi), color = 'green')

#Будуємо сепаратриси

separ1 = implicit\_plot(x, (x, -10, 10), (y, -10, 10), color = 'blue')

plot += separ1

separ2 = implicit\_plot(y, (x, -10, 10), (y, -10, 10), color = 'blue')

plot += separ2

#Будуємо ізокліну

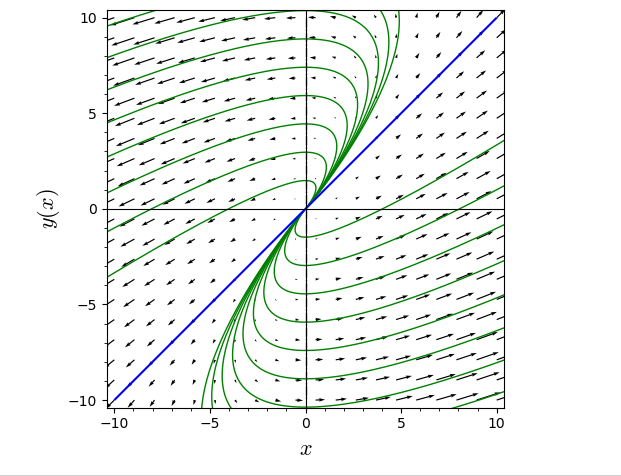
isocline = implicit\_plot(y-x/12, (x, -10, 10), (y, -10, 10), linewidth = 1, color = 'red', linestyle = 'dashdot')

plot += isocline

#Виводимо побудований фазовий портрет

plot.show(xmin = -10, xmax = 10, ymin = -10, ymax = 10, axes=true)

Розімкнена система



Замкнена система

