Національний Технічний Університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Навчально-Науковий Комплекс
«Інститут прикладного системного аналізу»

Лабороторна робота № 1

з дисципліни «Моделювання складних систем»

Тема: «Динаміка процесів, що описуються диференціальними рівняннями першого та другого порядку»

Виконали:
студенти гр. КА-41
Мельничук Валентин
Лочман Ярослава
Снігірьова Валерія

Прийняв:
професор кафедри ММСА,
д.т.н. Степашко В.С.

Київ 2018

Зміст

1 Опис та аналіз математичних моделей

1.1 Динаміка росту популяції (модель Фергюльста)

Описує динаміку росту популяції з урахуванням обмеженості ресурсів середовища:
$N' = \mu N(k - N) \mid N_0$
N – чисельність популяції
μ – коефіцієнти народжуваності (приріст популяції за одиницю часу)
k – ємність середовища (гранична чисельність популяції)
N_0 – початкова чисельність популяції
Стаціонарні точки:
$N' = 0 \Leftrightarrow \mu N(k - N) = 0$

$$\Rightarrow \begin{cases} N_{1,} = 0 \\ N_{2,} = k \end{cases}$$

Точка перегину:
$N'' = \mu N'(k - N) - \mu N N' = \mu N'((k - 2N) = \mu^2 N(k - N)(k - 2N)$
$N'' = 0 \Leftrightarrow \mu^2 N(k - N)(k - 2N) = 0$

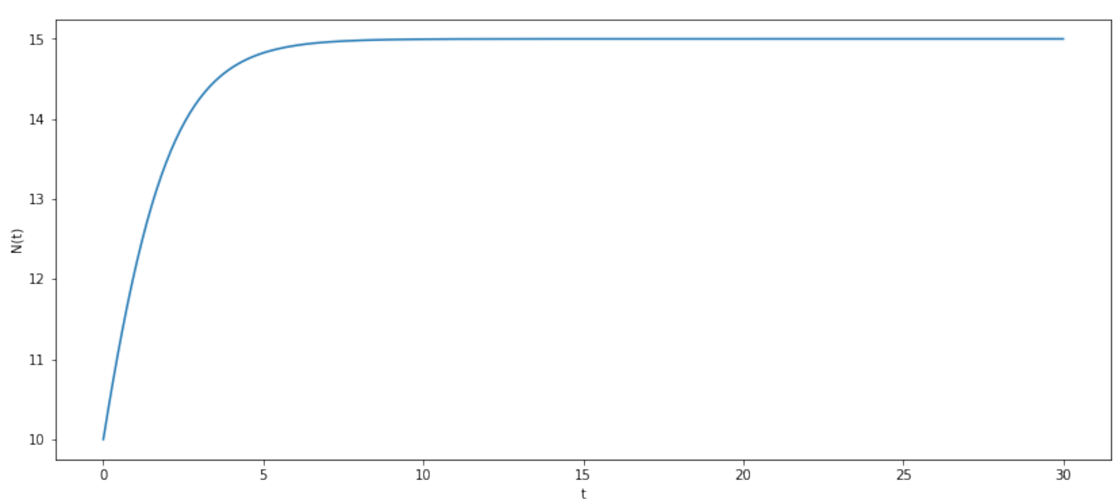
$$\Rightarrow \begin{cases} N_1 = 0 \\ N_2 = k/2 \\ N_3 = k \\ \text{немає сенсу розглядати } N < 0 \\ N = k \text{ - стаціонарна точка} \end{cases}$$

$\Rightarrow N_{\text{перегин}} = k/2$

1.1.1 Зростаючий процес

```
In [62]: model1(15, 0.05, 10)
```

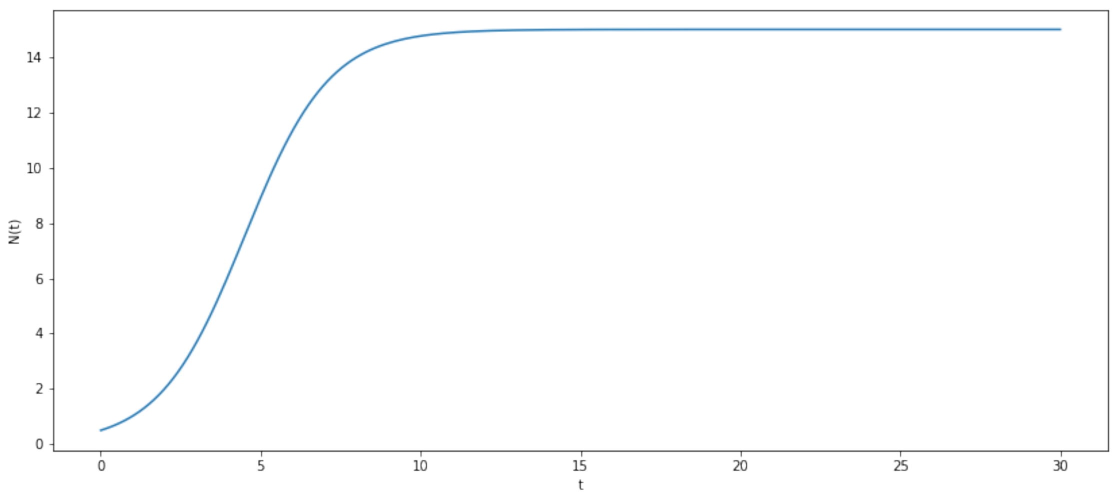
```
Ємність середовища (гранична чисельність): k = 15
Швидкість росту популяції: μ = 0.05
Початковий розмір популяції: N0 = 10
```



При $N0 < k$ та $\mu > 0$ відбуватиметься зростання популяції аж до граничної чисельності.

```
In [64]: model1(15, 0.05, 0.5)
```

```
Ємність середовища (гранична чисельність): k = 15
Швидкість росту популяції: μ = 0.05
Початковий розмір популяції: N0 = 0.5
```

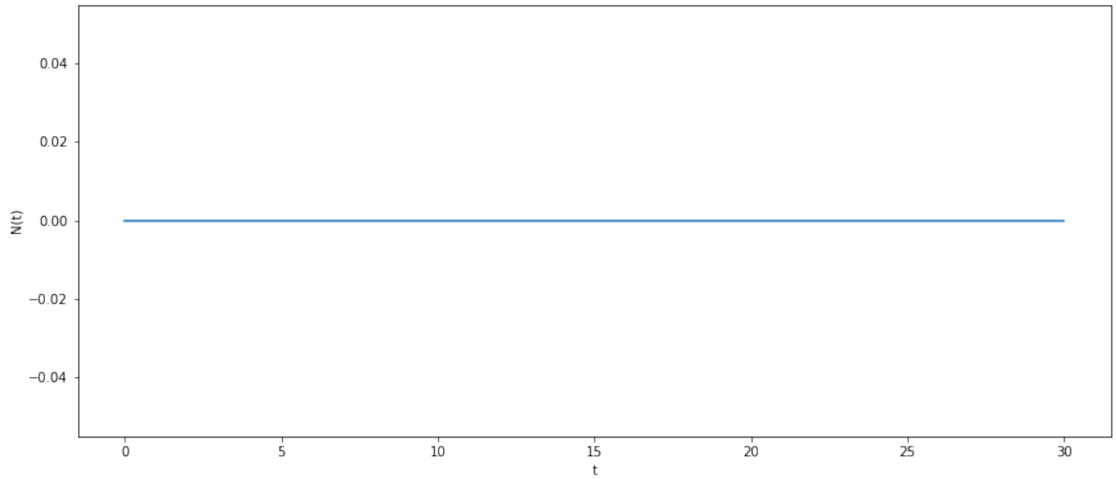


При цьому до значення чисельності $k/2$ швидкість зростання збільшується, а після нього зменшується до нуля (до чисельності популяції $N = k$) - дійсно, вище це було виведено аналітично, і підтверджено експериментально.

1.1.2 Стаціонарний процес

```
In [66]: model1(100, 0.05, 0)
```

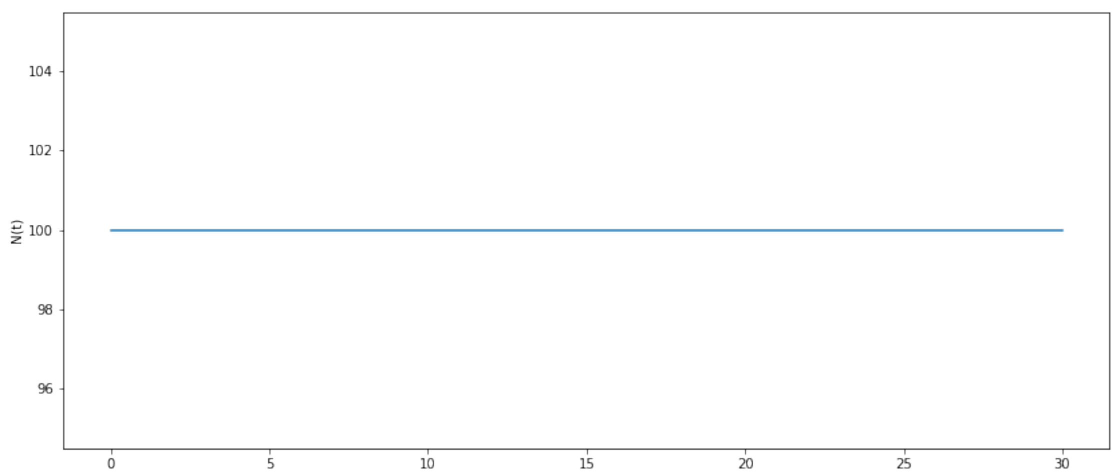
```
Ємність середовища (гранична чисельність): k = 100
Швидкість росту популяції: μ = 0.05
Початковий розмір популяції: N0 = 0
```



При початковій нульовій чисельності популяції з часом нічого не змінюється. Дійсно, розмноження популяції не відбувається через її відсутність.

```
In [67]: model1(100, 0.05, 100)
```

```
Ємність середовища (гранична чисельність): k = 100
Швидкість росту популяції: μ = 0.05
Початковий розмір популяції: N0 = 100
```

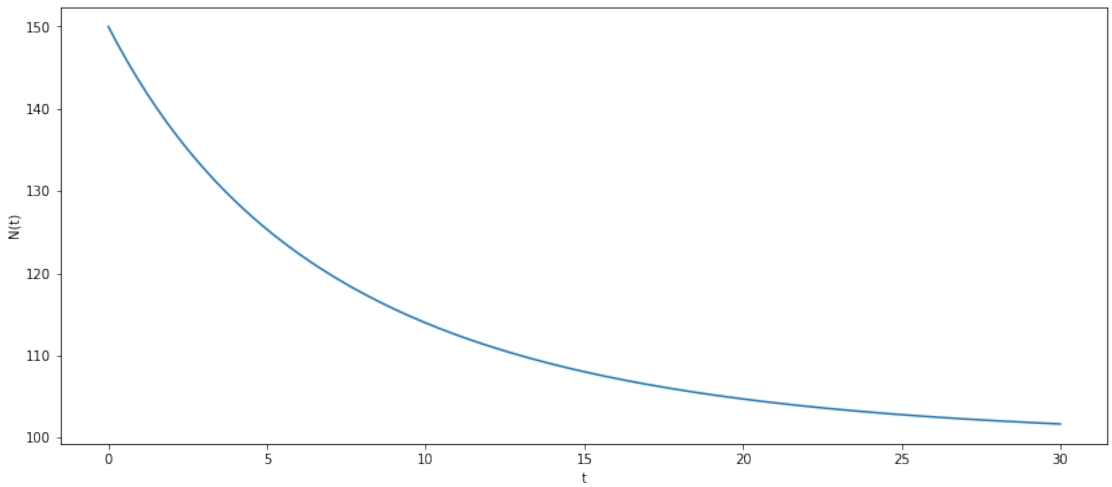


За початкової чисельності, рівної граничній чисельності, через обмеженість ресурсів популяція далі зростати не може, тож вона залишається такою й надалі.

1.1.3 Спадний процес

```
In [68]: model1(100, 0.001, 150)
```

```
Ємність середовища (гранична чисельність): k = 100
Швидкість росту популяції: μ = 0.001
Початковий розмір популяції: N0 = 150
```

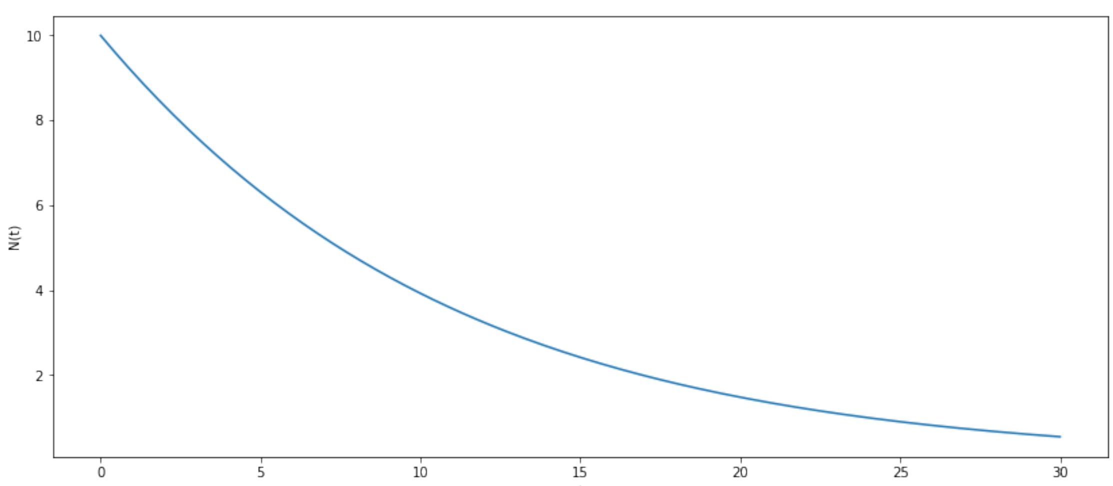


Коли $N_0 > k$, то спостерігається процес спадання N до значення k. Це так само пов'язано з обмеженістю ресурсів.

1.1.4 Повне вимирання

```
In [69]: model1(100, -0.001, 10)
```

```
Ємність середовища (гранична чисельність): k = 100
Швидкість росту популяції: μ = -0.001
Початковий розмір популяції: N0 = 10
```



Якщо швидкість росту популяції від'ємна ($\mu < 0$), тобто коли природна смертність більша за природну народжуваність, то при будь-якому $N_0 < k$ буде спостерігатися вимирання популяції (спадання N до нуля).

1.2 Односекторна модель економічної динаміки (модель Солоу)

Економічна система виробляє один продукт, який як споживається, так і інвестується. Експорт і імпорт не враховуються. Стан економіки в моделі Солоу визначається за допомогою п'яти ендогенних змінних, що змінюються з часом

$k' = sak^{\alpha} - (\mu + g)k$

a – рівень розвитку економіки

$0 < \alpha < 1$ – частка капіталу в продукції

$0 < \mu < 1$ – норма витрат на амортизацію

$0 < s < 1$ – норма накопичення

$0 < g < 1$ – темп приросту робочої сили

k_0 – початковий капітал

Стаціонарний розв'язок моделі економічної динаміки залежить від параметрів рівняння, початкових умов, а також конкретного варіанта виробничої функції. В цій роботі ми використовуємо виробничу функцію Кобба-Дугласа. Знайдемо для цього випадку стаціонарний розв'язок:

$k' = 0 \quad \Leftrightarrow \quad sak^{\alpha} - (\mu + g)k = 0$

Позначимо $\frac{sa}{\mu+g} = c$

$\Rightarrow ck^{\alpha} - k = 0$

Обидві функції (ck^{α} і k) є монотонними функціями, і перша - увігнута, а друга - опулка (і одночасно увігнута, до речі). Вони перетинаються в нулі та ще одній точці:

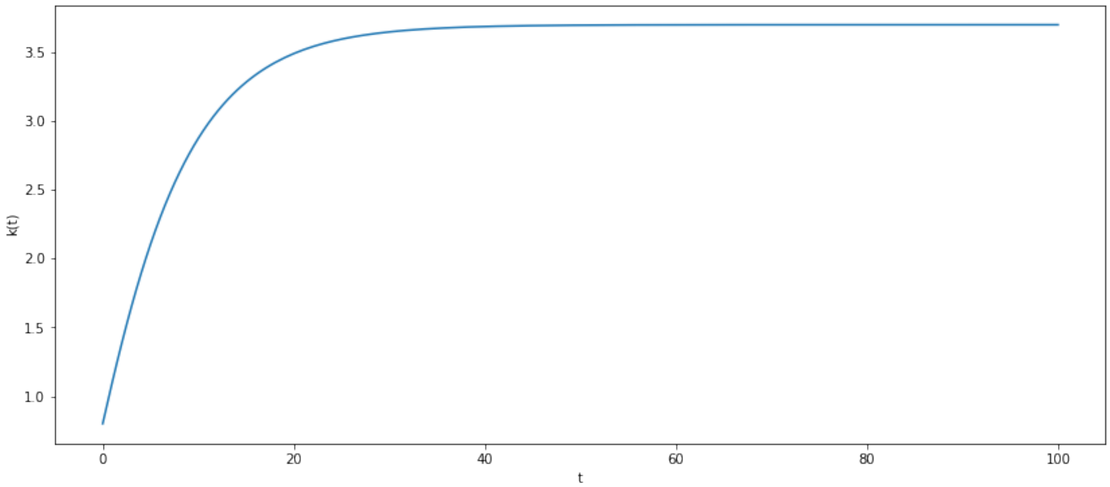
$ck^{\alpha-1} - 1 = 0 \quad \Rightarrow \quad k^{1-\alpha} = c \quad \Rightarrow \quad k = c^{\frac{1}{1-\alpha}}$

$$\Rightarrow \begin{cases} k_{1,} = 0 \\ k_{2,} = (\frac{sa}{\mu+g})^{\frac{1}{1-\alpha}} \end{cases}$$

1.2.1 Значення вхідних параметрів для України

```
In [71]: model2(0.2, 2.5, 0.3, 0.1, 0.1, 0.8)
```

```
Норма накопичення: s = 0.2
Рівень розвитку економіки: a = 2.5
Частка капіталу у сукупній продукції: α = 0.3
Норма витрат на амортизацію: μ = 0.1
Темп приросту робочої сили: q = 0.1
Початкова капіталозабезпеченість: k0 = 0.8
```

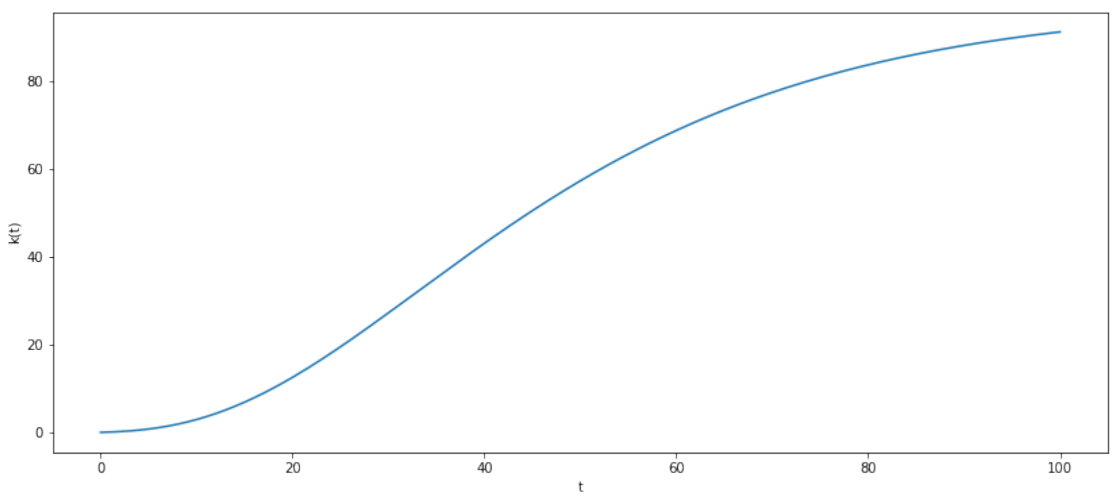


Процес не містить точки перегину - капіталоозброєність буде зростати, але швидкість зростання зменшуватиметься.

1.2.2 Зростаючий процес

```
In [72]: model2(0.2, 2.5, 0.8, 0.1, 0.1, 0.1)
```

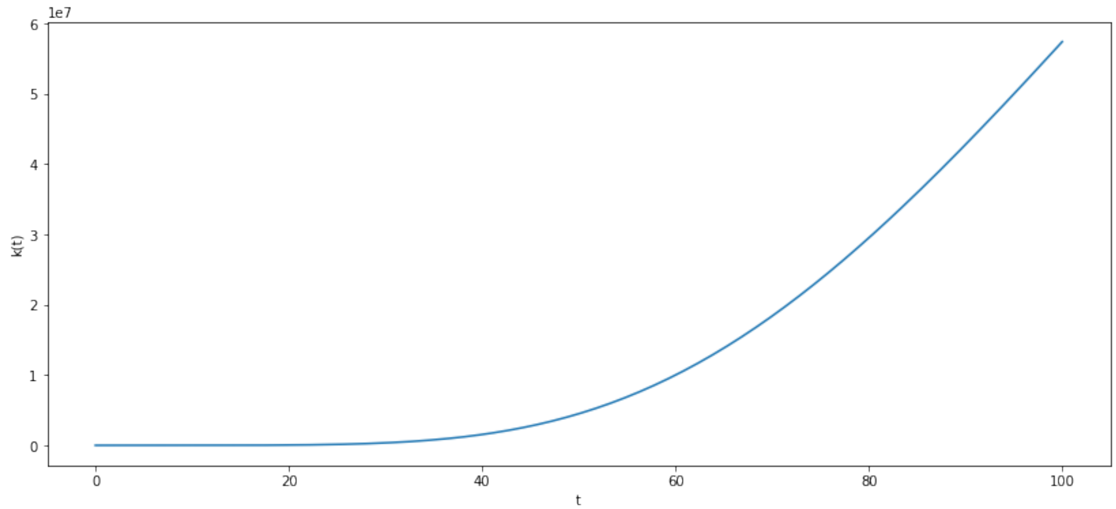
```
Норма накопичення: s = 0.2
Рівень розвитку економіки: a = 2.5
Частка капіталу у сукупній продукції: α = 0.8
Норма витрат на амортизацію: μ = 0.1
Темп приросту робочої сили: q = 0.1
Початкова капіталозабезпеченість: k0 = 0.1
```



Приклад зростаючого розв'язку, де наявна точка перегину, до якої відбувається пришвидшене зростання. На відміну від параметрів для України, в цьому процесі було збільшено лише частку капіталу у сукіпній продукції, що дало суттєві результати (капіталоозброєність при $t = 100$ зросла майже втричі).

```
In [73]: model2(0.5, 2.7, 0.9, 0.1, 0.1, 0.8)
```

Норма накопичення: s = 0.5
Рівень розвитку економіки: a = 2.7
Частка капіталу у сукупній продукції: α = 0.9
Норма витрат на амортизацію: μ = 0.1
Темп приросту робочої сили: q = 0.1
Початкова капіталозабезпеченість: k0 = 0.8

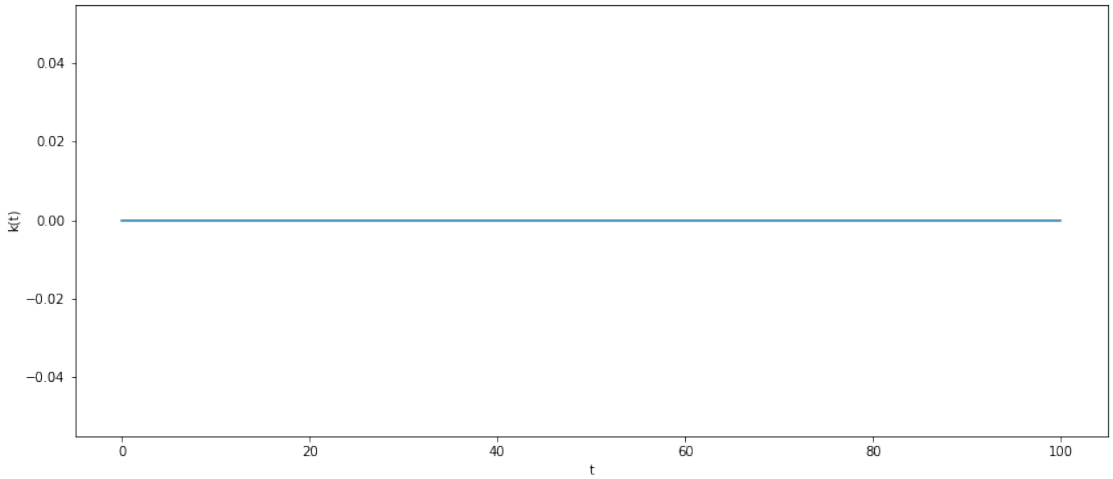


При збільшенні норми накопичення, рівня розвитку економіки або частки капіталу у сукупній продукції можна суттєво збільшити капіталоозброєність та швидкість її зростання.

1.2.3 Стаціонарний процес

```
In [74]: model2(0.2, 2.5, 0.3, 0.1, 0.1, 0)
```

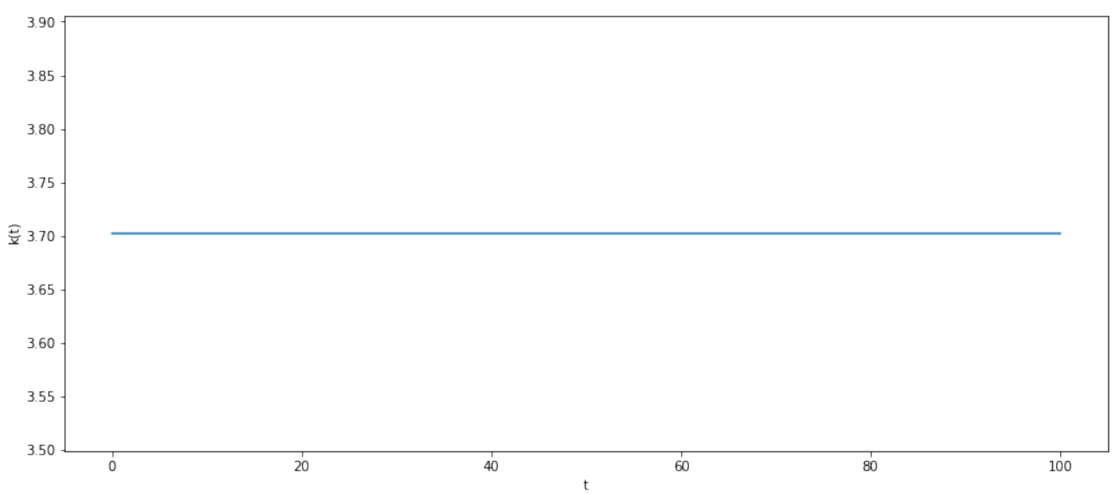
Норма накопичення: s = 0.2
Рівень розвитку економіки: a = 2.5
Частка капіталу у сукупній продукції: α = 0.3
Норма витрат на амортизацію: μ = 0.1
Темп приросту робочої сили: q = 0.1
Початкова капіталозабезпеченість: k0 = 0



Тривіальний стаціонарний розв'язок - $k_0 = 0$ (відсутність інвестицій)

```
In [75]: s = 0.2
         a = 2.5
         alpha = 0.3
         mu = 0.1
         g = 0.1
         k = (s * a / (mu + g)) ** (1 / (1 - alpha))
         model2(s, a, alpha, mu, g, k)
```

```
Норма накопичення: s = 0.2
Рівень розвитку економіки: a = 2.5
Частка капіталу у сукупній продукції: α = 0.3
Норма витрат на амортизацію: μ = 0.1
Темп приросту робочої сили: q = 0.1
Початкова капіталозабезпеченість: k0 = 3.7024203699314673
```

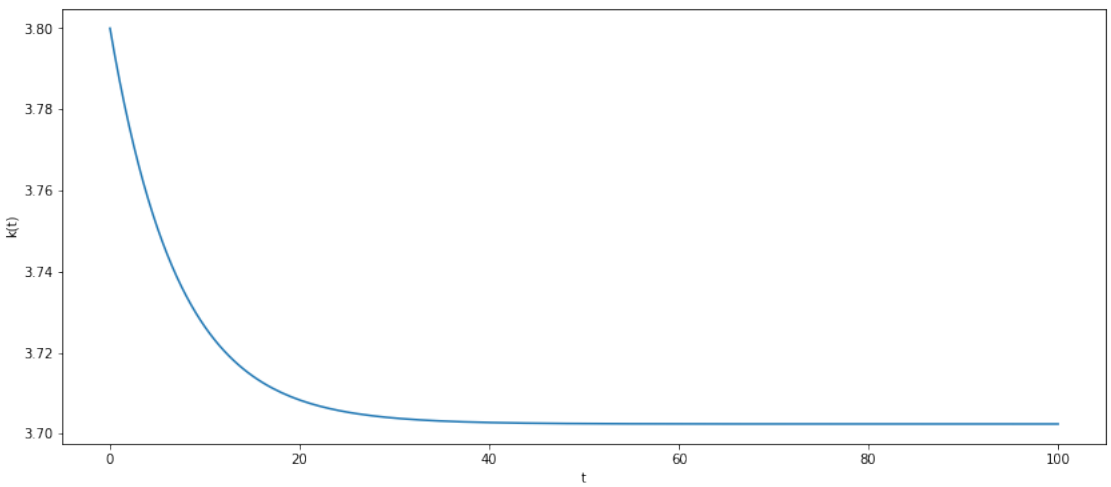


Стаціонарний розв'язок при $k_0 = (\frac{sa}{\mu+g})^{\frac{1}{1-\alpha}}$ - капіталоозброєність залишатиметься такою і надалі.

1.2.4 Спадний процес

```
In [76]: model2(0.2, 2.5, 0.3, 0.1, 0.1, 3.8)
```

```
Норма накопичення: s = 0.2
Рівень розвитку економіки: a = 2.5
Частка капіталу у сукупній продукції: α = 0.3
Норма витрат на амортизацію: μ = 0.1
Темп приросту робочої сили: q = 0.1
Початкова капіталозабезпеченість: k0 = 3.8
```



Коли початкова капіталозброєність більша, ніж стаціонарний стан ($k_0 > (\frac{sa}{\mu+g})^{\frac{1}{1-\alpha}}$), спостерігається явище "проїдання фондів коли при недостатньо високому рівні економіки і невеликій частці капіталу у сукупній продукції вкладають багато інвестицій.

1.3 Вимушені коливання

$x'' + 2\delta x' + \omega_0^2 x = 0$
δ - коефіцієнт згасання
ω_0 - власна частота коливань
ω – частота коливань зовнішньої сили
f_0 – амплітуда зовнішньої сили

$$\begin{cases} \ddot{x}(t) + 2\delta\dot{x}(t) + \omega_0^2 x(t) = 0 \\ x(t) = Acos(\omega t + \phi);\ \ \dot{x}(t) = -A\omega sin(\omega t + \phi);\ \ \ddot{x}(t) = -A\omega^2 cos(\omega t + \phi) \end{cases}$$

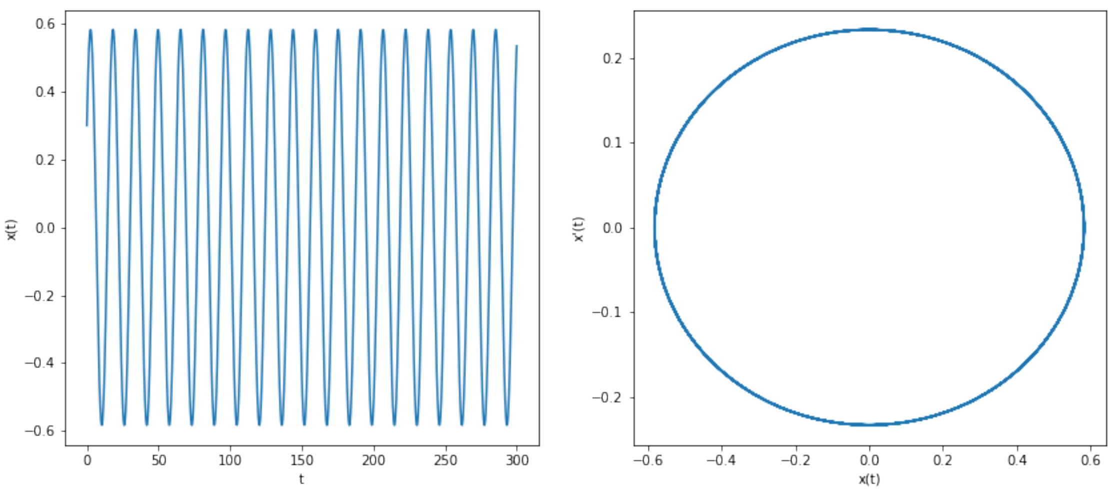
$$\Rightarrow \begin{cases} A = \frac{f_0}{\sqrt{(\omega_0^2-\omega^2)^2+4\delta^2\omega^2}} \\ tg(\phi) = -\frac{2\delta\omega}{\omega_0^2-\omega^2)} \end{cases}$$

$x(t) = \frac{f_0}{\sqrt{(\omega_0^2-\omega^2)^2+4\delta^2\omega^2}} cos(\omega t + \phi)$, де $tg(\phi) = -\frac{2\delta\omega}{\omega_0^2-\omega^2)}$

1.3.1 Гармонічні коливання

```
In [77]: model3(0, 0.4, 0, 0, 0.3, 0.2)
```

```
Коефіцієнт згасання: δ = 0
Власна частота: ω0 = 0.4
Частота зовнішньої сили: ω = 0
Амплітуда зовнішньої сили: f0 = 0
Початкове положення: x0 = 0.3
Початкова швидкіть: x0' = 0.2
```

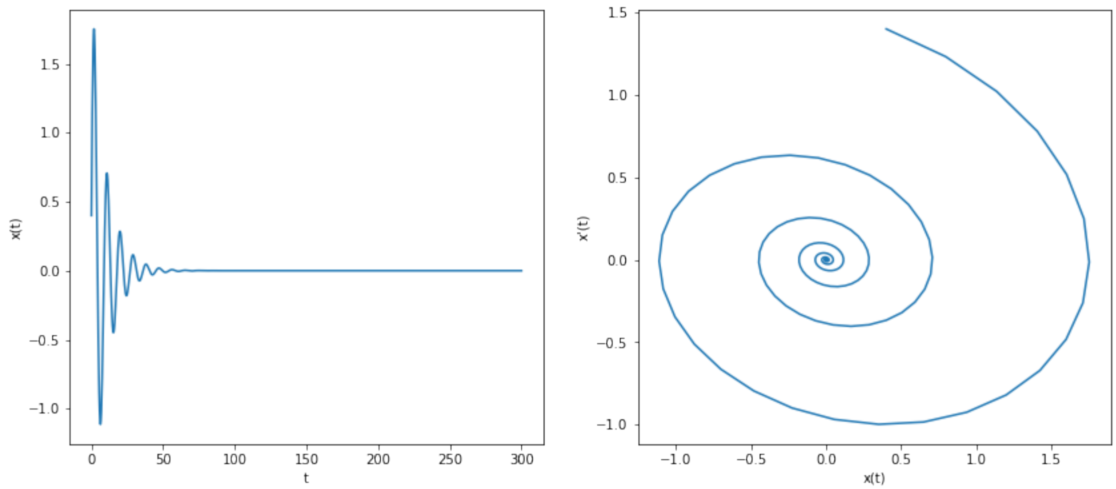


Графік гармонічних коливань (коли $\delta = 0$, $f_0 = 0$). Амплітуда не міняється з часом, немає ніяких зовнішніх сил.

1.3.2 Згасаючі коливання

```
In [78]: model3(0.1, 0.7, 0, 0, 0.4, 1.4)
```

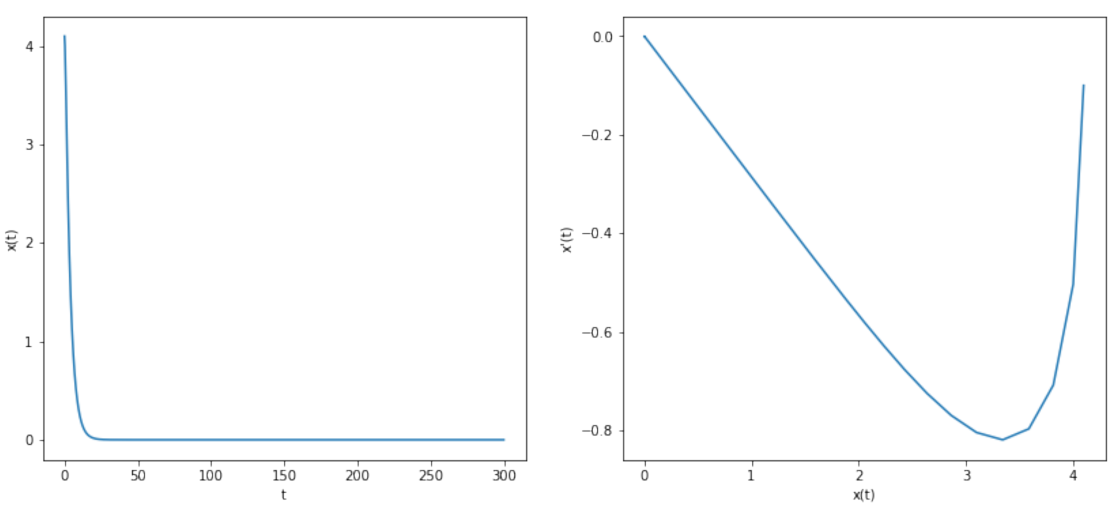
```
Коефіцієнт згасання: δ = 0.1
Власна частота: ω0 = 0.7
Частота зовнішньої сили: ω = 0
Амплітуда зовнішньої сили: f0 = 0
Початкове положення: x0 = 0.4
Початкова швидкіть: x0' = 1.4
```



Графік згасаючих коливань, коли коефіцієнт згасання $0 < \delta < 1$ (енергія коливань зменшується з плином часу, наприклад, під дією сил опору середовища). Зовнішня сила відсутня $f_0 = 0$. Амплітуда коливань в цьому випадку спадає експоненційно: $A(t) = Ce^{-\zeta\delta_0 t}$, де C залежить від початкових умов.

```
In [79]: model3(1, 0.7, 0, 0, 4.1, -0.1)
```

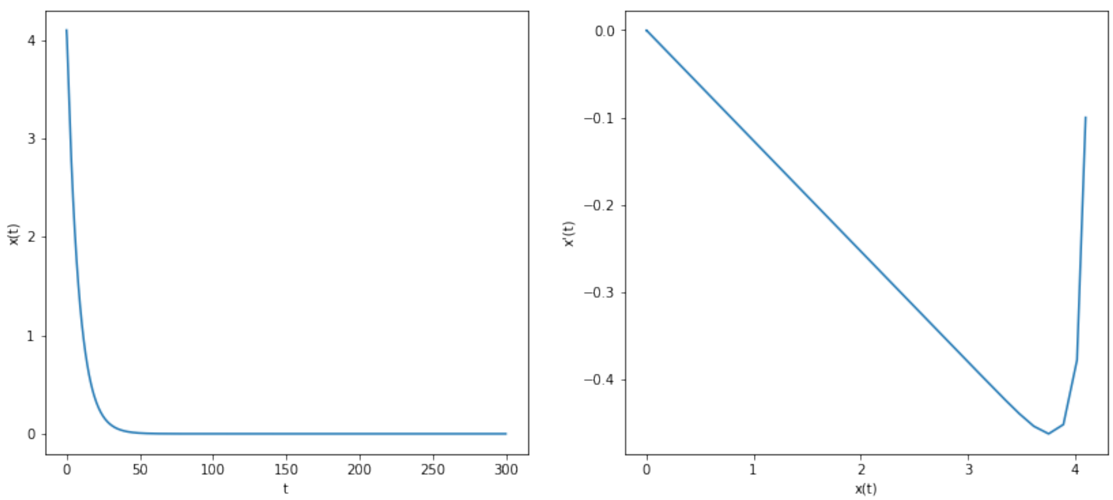
```
Коефіцієнт згасання: δ = 1
Власна частота: ω0 = 0.7
Частота зовнішньої сили: ω = 0
Амплітуда зовнішньої сили: f0 = 0
Початкове положення: x0 = 4.1
Початкова швидкіть: x0' = -0.1
```



Вище зображено графік згасаючих коливань, коли коефіцієнт згасання $\delta = 1$, а зовнішня сила відсутня $f_0 = 0$. При цьому коливання відсутні і розв'язок спадає відповідно до наступного закону: $x(t) = (c_1 t + c_2)e^{-\omega_o t}$, де c_1 і c_2 знаходяться через початкові умови.

```
In [80]: model3(2, 0.7, 0, 0, 4.1, -0.1)
```

```
Коефіцієнт згасання: δ = 2
Власна частота: ω0 = 0.7
Частота зовнішньої сили: ω = 0
Амплітуда зовнішньої сили: f0 = 0
Початкове положення: x0 = 4.1
Початкова швидкість: x0' = -0.1
```

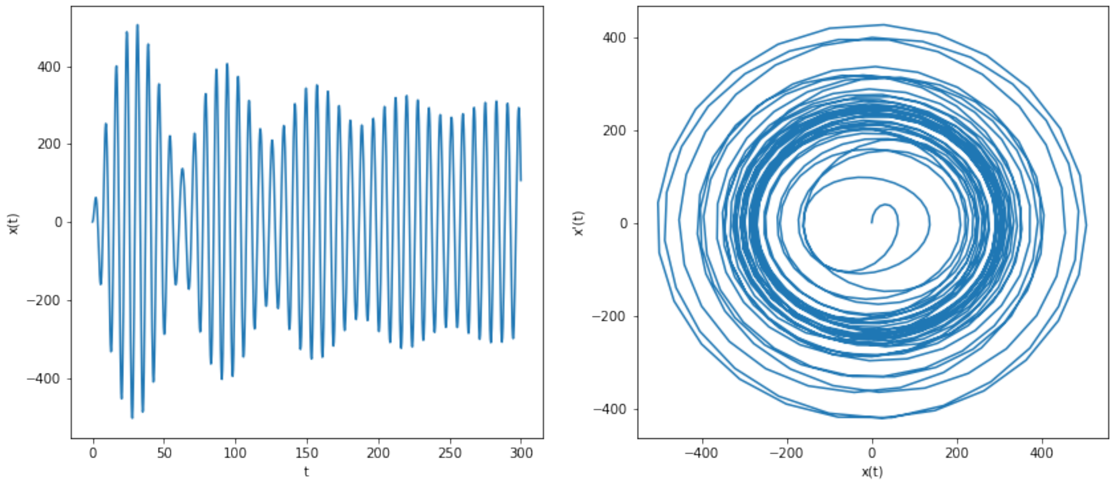


Вище зображено графік згасаючих коливань, коли коефіцієнт згасання $\delta > 1$, а зовнішня сила відсутня $f_0 = 0$. При цьому коливання відсутні і розв'язок експоненційно спадає відповідно до наступного закону: $x(t) = c_1 e^{\lambda_- t} + c_2 e^{\lambda_+ t}$, де c_1 і c_2 знаходяться через початкові умови.

1.3.3 Вимушені коливання

```
In [81]: model3(0.01, 0.9, 0.8, 50, 0.4, 1.4)
```

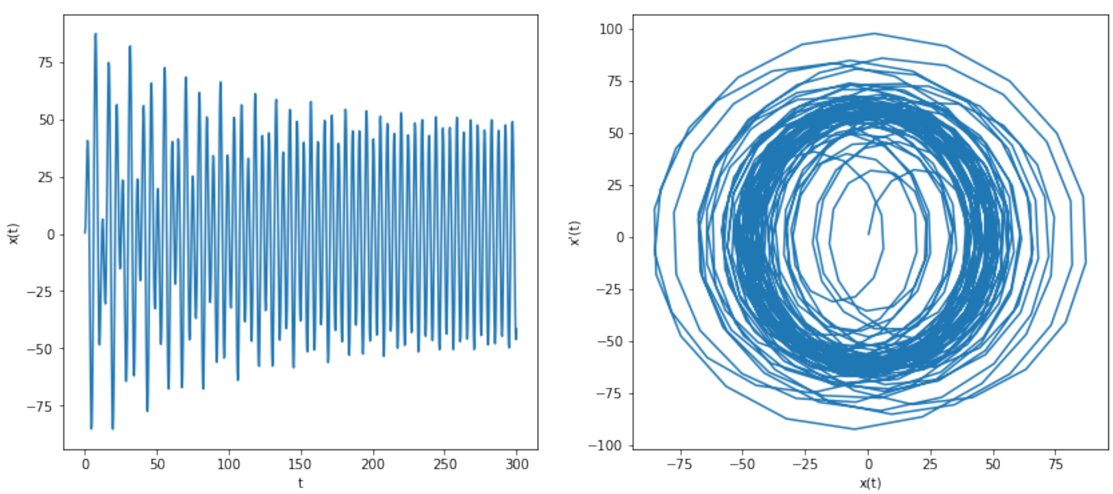
```
Коефіцієнт згасання: δ = 0.01
Власна частота: ω0 = 0.9
Частота зовнішньої сили: ω = 0.8
Амплітуда зовнішньої сили: f0 = 50
Початкове положення: x0 = 0.4
Початкова швидкість: x0' = 1.4
```



Вимушені коливання при відсутності резонансу, графік розв'язку містить дві частоти - зовнішню і внутрішню. Вище зображено випадок, коли $\omega_0 > \omega$

```
In [82]: model3(0.01, 0.8, 1.3, 50, 0.4, 1.4)
```

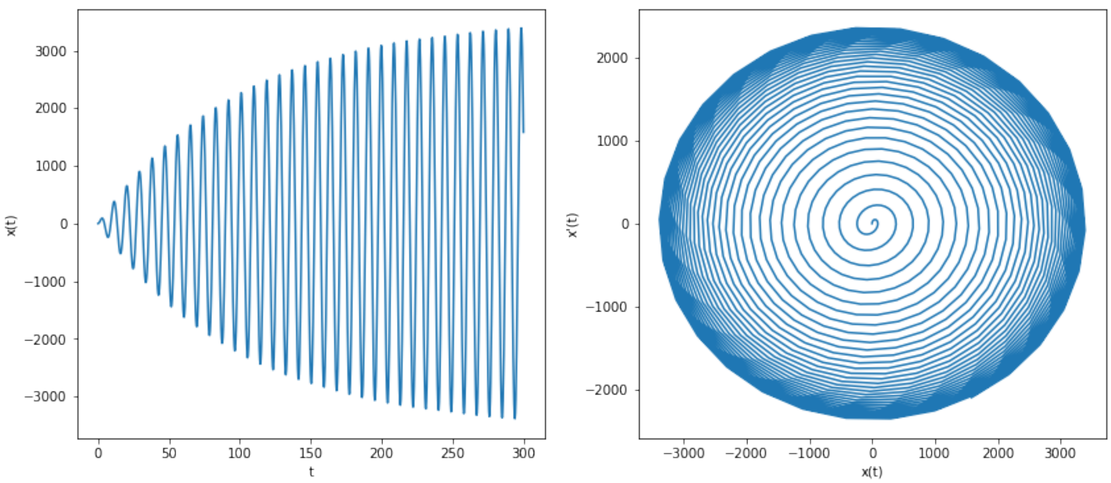
```
Коефіцієнт згасання: δ = 0.01
Власна частота: ω0 = 0.8
Частота зовнішньої сили: ω = 1.3
Амплітуда зовнішньої сили: f0 = 50
Початкове положення: x0 = 0.4
Початкова швидкіть: x0' = 1.4
```



Вимушені коливання, коли $\omega_0 < \omega$

```
In [83]: model3(0.01, 0.7, 0.7, 50, 0.4, 1.4)
```

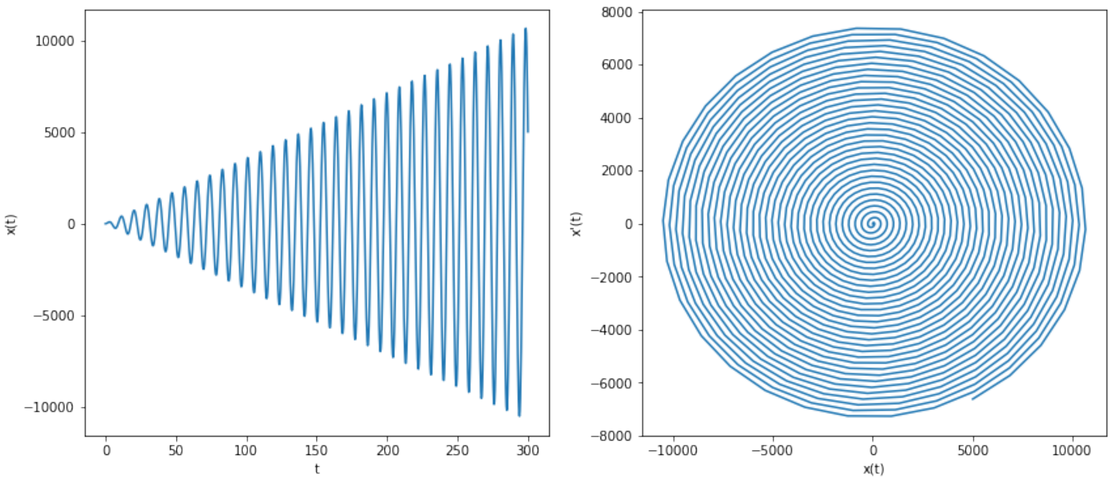
```
Коефіцієнт згасання: δ = 0.01
Власна частота: ω0 = 0.7
Частота зовнішньої сили: ω = 0.7
Амплітуда зовнішньої сили: f0 = 50
Початкове положення: x0 = 0.4
Початкова швидкіть: x0' = 1.4
```



Явище резонансу при $\delta > 0$. Причому, максимальна амплітуда $A_{max} \approx \frac{\omega_0 f_0}{\delta}$

```
In [84]: model3(0., 0.7, 0.7, 50, 0.4, 1.4)
```

```
Коефіцієнт згасання: δ = 0.0
Власна частота: ω0 = 0.7
Частота зовнішньої сили: ω = 0.7
Амплітуда зовнішньої сили: f0 = 50
Початкове положення: x0 = 0.4
Початкова швидкіть: x0' = 1.4
```



Явище резонансу при $\delta = 0$, амплітуда нескінченно зростає.

1.4 Коливання у системі «хижак-жертва»

$$\begin{cases} x' = (\alpha_x y - \beta_x)x & - \text{ для популяції хижаків} \\ y' = (\alpha_y - \beta_y x)y & - \text{ для популяції жертв} \end{cases}$$

α_x - "норма споживання"жертв
α_y - природна народжуваність жертв
β_x - природна смертність хижаків
β_y - "норма споживаності"жертв
x_0 - початкова кількість хижаків
y_0 - початкова кількість жертв

Характерною особливістю рівннянь є те, що їхнім розв'язком є автоколивання, тобто амплітуда і період коливань залежать від властивостей самої системи і не залежать від початкових умов, далі це покажемо. Тут відбуваються такі процеси: розмноження жертв та їхня гибель в результаті поїдання хижаками, розмноження та вимирання хижаків.

Стаціонарні точки:

$$\begin{cases} x' = 0 \\ y' = 0 \end{cases} \Leftrightarrow \begin{cases} (\alpha_x y - \beta_x)x = 0 \\ (\alpha_y - \beta_y x)y = 0 \end{cases} \Rightarrow \left[\begin{array}{l} \begin{cases} x^* = 0 \\ y^* = 0 \end{cases} \\ \begin{cases} x^* = \frac{\alpha_y}{\beta_y} \\ y^* = \frac{\beta_x}{\alpha_x} \end{cases} \end{array} \right.$$

Стаціонарна точка $(x^*, y^*) = (\frac{\alpha_y}{\beta_y}, \frac{\beta_x}{\alpha_x})$ $-$ положення рівноваги, в якому чисельність жертв і хижаків залишаються сталими. З відхиленням від цієї точки спостерігаються коливальні процеси зміни популяцій.

Коливання:

Дослідимо систему при незначних відхиленнях від (x^*, y^*):

Нехай $x = x^* + u, y = y^* + v;$ u, v - відносно маленькі величини
$\dot{u} = \dot{x} = (\alpha_x v + \alpha_x y^* - \beta_x)(x^* + u) = \alpha_x v(x^* + u) \approx \alpha_x v x^* = \frac{\alpha_x \alpha_y}{\beta_y} v$

$\dot{v} = \dot{x} = (\alpha_y - \beta_y x^* - \beta_y u)(y^* + v) = (-\beta_y u)(y^* + v) \approx -\beta_y u y^* = -\frac{\beta_x \beta_y}{\alpha_x} u$

$\Rightarrow \quad \ddot{u} = \frac{\alpha_x \alpha_y}{\beta_y} \dot{v} = -\alpha_y \beta_x u \quad \Rightarrow \quad \ddot{u} + \alpha_y \beta_x u = 0$

$\ddot{v} = -\frac{\beta_x \beta_y}{\alpha_x} \dot{u} = -\alpha_y \beta_x v \quad \Rightarrow \quad \ddot{v} + \alpha_y \beta_x v = 0$

$\Rightarrow \omega_0 = \sqrt{\alpha_y \beta_x}$ як для u, так і для v

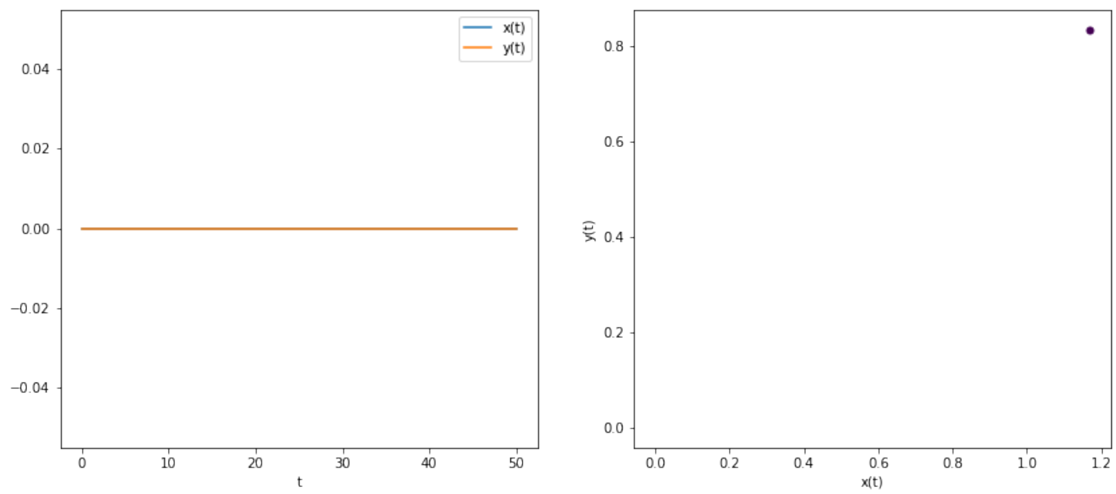
Отже, частота коливань для обох популяцій однакова і визначається властивостями системи - природною народжуваністю жертв і природною смертністю хижаків

Амплітуди і фази коливань залежать від початкових умов.

1.4.1 Стаціонарний процес

```
In [86]: model4(0.6, 0.7, 0.5, 0.6, 0, 0)
```

```
"Норма споживання" жертв: αx = 0.6
Природна народжуваність жертв: αy = 0.7
Природна смертність хижаків: βx = 0.5
"Норма споживаності" жертв: βy = 0.6
Початкова кількість хижаків: x0 = 0
Початкова кількість жертв: y0 = 0
```

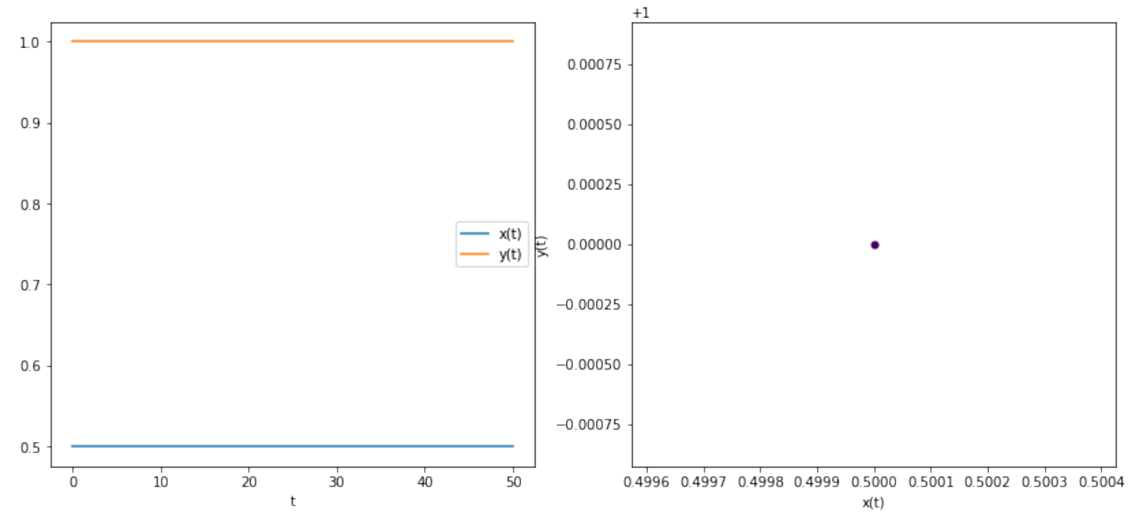


Тривіальний стаціонарний розв'язок: $\begin{cases} x_0 = 0 \\ y_0 = 0 \end{cases}$ тобто в популяції немає ні жертв, ні хижаків.

```
In [87]: αx = 0.3
         αy = 0.3
         βx = 0.3
         βy = 0.6
         model4(αx, αy, βx, βy, αy/βy, βx/αx)
```

```
"Норма споживання" жертв: αx = 0.3
Природна народжуваність жертв: αy = 0.3
```

```
Природна смертність хижаків: βx = 0.3
"Норма споживаності" жертв: βy = 0.6
Початкова кількість хижаків: x0 = 0.5
Початкова кількість жертв: y0 = 1.0
```



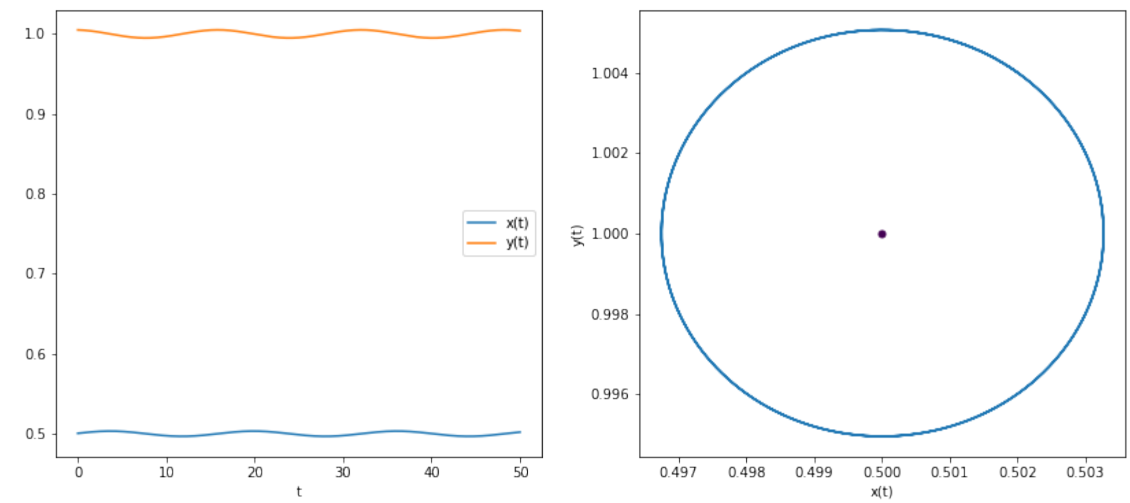
Стаціонарний розв'язок, коли чисельності популяцій не змінюються:

$$\begin{cases} x_0 = \frac{\alpha_y}{\beta_y} \\ y_0 = \frac{\beta_x}{\alpha_x} \end{cases}$$

1.4.2 Коливання при незначних відхиленнях початкових умов від рівноваги

```
In [88]: model4(0.5, 0.3, 0.5, 0.6, 0.5005, 1.005)
```

```
"Норма споживання" жертв: αx = 0.5
Природна народжуваність жертв: αy = 0.3
Природна смертність хижаків: βx = 0.5
"Норма споживаності" жертв: βy = 0.6
Початкова кількість хижаків: x0 = 0.5005
Початкова кількість жертв: y0 = 1.005
```

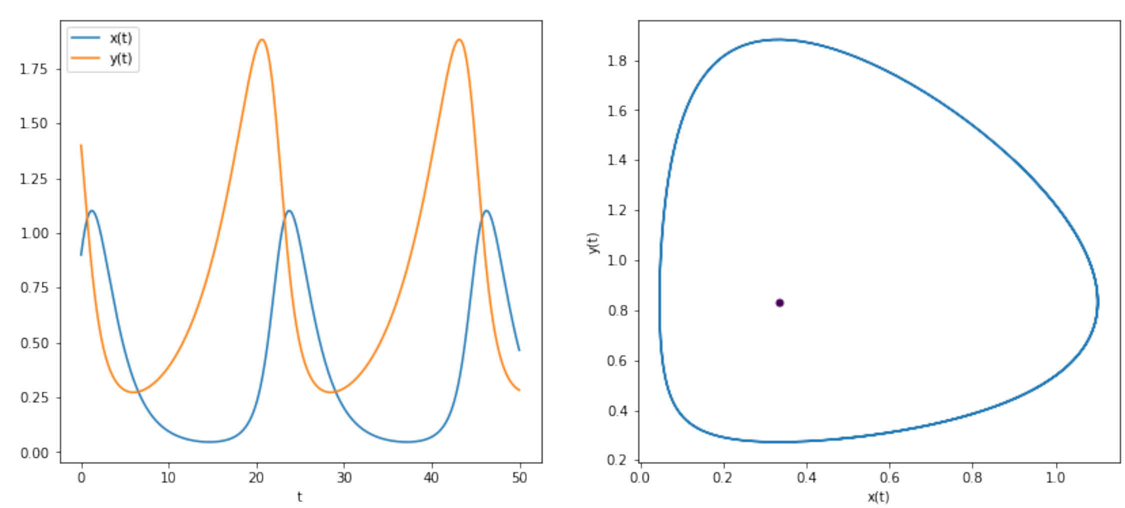


Розв'язком завжди буде періодична функція. Фазові траєкторії — замкнуті криві, всередині яких знаходиться фокус (точка рівноваги), яку позначено фіолетовим. При незначних відхиленнях фігура схожа на еліпс, але не є еліпсом (при відхиленнях, прямуючих до нуля, фігура прямуватиме до еліпса)

1.4.3 Коливання при значних відхиленнях початкових умов від рівноваги

```
In [89]: model4(0.6, 0.2, 0.5, 0.6, 0.9, 1.4)
```

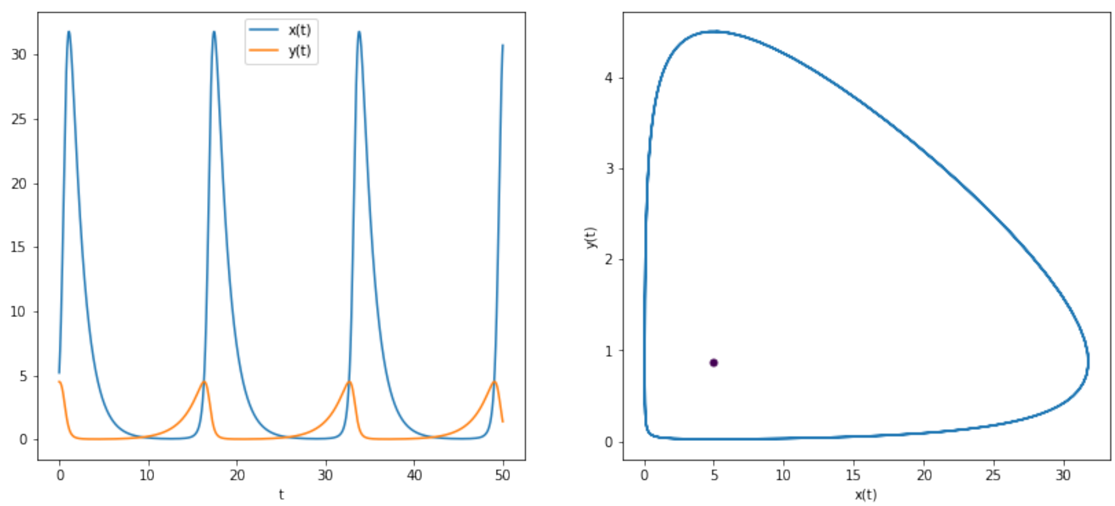
```
"Норма споживання" жертв: αx = 0.6
Природна народжуваність жертв: αy = 0.2
Природна смертність хижаків: βx = 0.5
"Норма споживаності" жертв: βy = 0.6
Початкова кількість хижаків: x0 = 0.9
Початкова кількість жертв: y0 = 1.4
```



Можна помітити, що кількість хижаків - періодично змінються із запізненням відносно до кількості жертв. Це підтверджує той факт, що розмноження хижаків залежить прямо пропорційно від кількості іжі, тобто, кількості потенційних жертв у популяції. Дійсно, при значному розмноженні жертв створюються умови для розмноження хижаків завдяки доступності їжі. Але розмноження хижаків призводить до зменшення числа жертв. Коли число жертв сильно падає, хижаки теж гинуть через недостатню кількість їжі. Тільки тоді, коли кількість хижаків досягає мінімуму, популяція жертв знову починає зростати.

```
In [90]: model4(0.8, 0.5, 0.7, 0.1, 5.2, 4.5)
```

```
"Норма споживання" жертв: αx = 0.8
Природна народжуваність жертв: αy = 0.5
Природна смертність хижаків: βx = 0.7
"Норма споживаності" жертв: βy = 0.1
Початкова кількість хижаків: x0 = 5.2
Початкова кількість жертв: y0 = 4.5
```



В даному випадку кількість хижаків набагато більша. На це дуже впливає норма споживання жертв і норма споживаності жертв - тобто те, скільки потенційно може з'явитися хижаків від поїдання однієї жертви, та те, скільки жертв потенційно вимре від одного хижака відповідно. В даному випадку перший параметр досить високий, а другий досить низький. Також слід відмітити досить високу народжуваність жертв. Це все призводить до такої картини.

2 Висновки

В даній роботі ми дослідили чотири процеси, аналітично та експериментально знайшли в них стаціонарні точки, точки перегину та інші різні характеристики типу амплітуди, частоти коливань; проілюстрували та проаналізували важливі типи розв'язків.

3 Код програми

In [70]:
```
from ipywidgets import *
import numpy as np
from scipy.integrate import odeint
import matplotlib.pyplot as plt
from IPython.display import display
%matplotlib inline

from pylab import rcParams
rcParams['figure.figsize'] = 14, 6
style = {'description_width': 'initial'}
layout = Layout(width = '400px')
models = [None] * 4



def model1(k, μ, N0):
    def model1_eq(N,t):
        return μ*N*(k-N)

    t = np.linspace(0, 30, num=200)

    # solve ODE
    N = odeint(model1_eq, N0, t)

    print('Ємність середовища (гранична чисельність): k = {}'.format(k))
    print('Швидкість росту популяції: μ = {}'.format(μ))
    print('Початковий розмір популяції: N0 = {}'.format(N0))

    # plot results
    plt.plot(t, N)
    plt.xlabel('t')
    plt.ylabel('N(t)')
    plt.show()

def plot_model1(k, mu, N0):
    description1 = Label(value="$N'=μN(k-N)$")
    k_val = FloatText(value=k, description = \
    'Ємність середовища (гранична чисельність): $k = $', step=0.1,
                      style = style, layout = layout)
    μ_val = FloatText(value=mu, description = \
    'Швидкість росту популяції: $μ = $', step=0.1,
                      style = style, layout = layout)
    N0_val = FloatText(value=N0, description = \
    'Початковий розмір популяції: $N_0 = $', step=0.1,
                       style = style, layout = layout)
    models[0] = VBox([description1, interactive(model1, k=k_val,
```

```
                                        μ=μ_val, N0=N0_val,
                                        continuous_update=True)])
    display(models[0])


def model2(s, a, α, μ, q, k0):
    def model2_eq(k,t):
        return s*a*k**α-(μ+q)*k

    t = np.linspace(0, 100, num=200)

    # solve ODE
    k = odeint(model2_eq, k0, t)

    print('Норма накопичення: s = {}'.format(s))
    print('Рівень розвитку економіки: a = {}'.format(a))
    print('Частка капіталу у сукупній продукції: α = {}'.format(α))
    print('Норма витрат на амортизацію: μ = {}'.format(μ))
    print('Темп приросту робочої сили: q = {}'.format(q))
    print('Початкова капіталозабезпеченість: k0 = {}'.format(k0))

    # plot results
    plt.plot(t, k)
    plt.xlabel('t')
    plt.ylabel('k(t)')
    plt.show()

def plot_model2(s, a, alpha, mu, q, k0):
    description2 = Label(value="$k'=sak^α-(μ+q)k$")
    s_val = FloatText(s, description = \
    'Норма накопичення: $s = $', step=0.1, style = style, layout = layout)
    a_val = FloatText(a, description = \
    'Рівень розвитку економіки: $a = $', step=0.1,
                  style = style, layout = layout)
    α_val = FloatText(alpha, description = \
    'Частка капіталу у сукупній продукції: $α = $', step=0.1,
                  style = style, layout = layout)
    μ_val = FloatText(mu, description = \
    'Норма витрат на амортизацію: $μ = $', step=0.1,
                  style = style, layout = layout)
    q_val = FloatText(q, description = \
    'Темп приросту робочої сили: $q = $', step=0.1,
                  style = style, layout = layout)
    k0_val = FloatText(k0, description = \
    'Початкова капіталозабезпеченість: $k_0 = $', step=0.1,
                   style = style, layout = layout)
    models[1] = VBox([description2, interactive(model2, s=s_val,
                                        a=a_val, α=α_val,
```

```
                                                  μ=μ_val, q=q_val, k0=k0_val,
                                                  continuous_update=True)])
    display(models[1])


def model3(δ, ω0, ω, f0, x0, x00):

    #x_0' = x_1 = x'
    #x_1' = x'' = f0 * np.cos(ω * t) - 2 * δ * x[1] - (ω0 ** 2) * x[0]
    def model3_eq(x,t):
        return [x[1], f0 * np.cos(ω * t) - 2 * δ * x[1] - (ω0 ** 2) * x[0]]

    t = np.linspace(0, 300, num=1000)

    # solve ODE
    x = odeint(model3_eq, np.array([x0, x00]), t)

    print('Коефіцієнт згасання: δ = {}'.format(δ))
    print('Власна частота: ω0 = {}'.format(ω0))
    print('Частота зовнішньої сили: ω = {}'.format(ω))
    print('Амплітуда зовнішньої сили: f0 = {}'.format(f0))
    print('Початкове положення: x0 = {}'.format(x0))
    print("Початкова швидкіть: x0' = {}".format(x00))

    # plot results
    fig, ax = plt.subplots(nrows=1, ncols=2)
    ax[0].plot(t, x[:,0])
    ax[0].set_xlabel('t')
    ax[0].set_ylabel('x(t)')
    ax[1].plot(x[:,0], x[:,1])
    ax[1].set_xlabel('x(t)')
    ax[1].set_ylabel("x'(t)")
    plt.show()

def plot_model3(delta, omega0, omega, f0, x0, x00):
    description3 = Label(value="$x''+2δ x'+ω_0^2 x=f_0 cos(ωt)$")
    δ_val = FloatText(value=delta, description = \
    'Коефіцієнт згасання: $δ = $', step=0.1, style = style, layout = layout)
    ω0_val = FloatText(value=omega0, description = \
    'Власна частота: $ω_0 = $', step=0.1, style = style, layout = layout)
    ω_val = FloatText(value=omega, description = \
    'Частота зовнішньої сили: $ω = $', step=0.1,
                      style = style, layout = layout)
    f0_val = FloatText(value=f0, description = \
    'Амплітуда зовнішньої сили: $f_0 = $', step=0.1,
                       style = style, layout = layout)
    x0_val = FloatText(value=x0, description = \
    'Початкове положення: $x_0 = $', step=0.1,
```

```
                          style = style, layout = layout)
    x00_val = FloatText(value=x00, description = \
    "Початкова швидкіть: $x_0' = $", step=0.1,
                           style = style, layout = layout)
    models[2] = VBox([description3, interactive(model3, δ=δ_val, ω0=ω0_val,
                                                ω=ω_val, f0=f0_val,
                                                x0=x0_val, x00=x00_val,
                                                continuous_update=True)])
    display(models[2])


def model4(αx, αy, βx, βy, x0, y0):
    def model4_eq(x,t):
        return [(αx * x[1] - βx) * x[0], (αy - x[0] * βy) * x[1]]

    t = np.linspace(0, 50, num=1000)

    # solve ODE
    x = odeint(model4_eq, np.array([x0, y0]), t)

    print('"Норма споживання" жертв: αx = {}'.format(αx))
    print('Природна народжуваність жертв: αy = {}'.format(αy))
    print('Природна смертність хижаків: βx = {}'.format(βx))
    print('"Норма споживаності" жертв: βy = {}'.format(βy))
    print('Початкова кількість хижаків: x0 = {}'.format(x0))
    print("Початкова кількість жертв: y0 = {}".format(y0))

    # plot results
    fig, ax = plt.subplots(nrows=1, ncols=2)
    ax[0].plot(t, x[:,0], label = 'x(t)')
    ax[0].plot(t, x[:,1], label = 'y(t)')
    ax[0].legend()
    ax[0].set_xlabel('t')
    ax[1].plot(x[:,0], x[:,1])
    if βy != 0 and αx != 0:
        ax[1].scatter(αy/βy, βx/αx, s=25, c=4)
    ax[1].set_xlabel('x(t)')
    ax[1].set_ylabel("y(t)")
    plt.show()

def plot_model4(alpha_x, alpha_y, beta_x, beta_y, x0, y0):
    description4 = Label(value=\
    r"""$ \bigg\{ \matrix{x'=(α_x y - β_x)x \cr y'=(α_y - β_y x)y} $""",
                            layout = Layout(height = '50px'))
    αx_val = FloatText(value=alpha_x, description = \
   '"Норма споживання" жертв: $α_x = $', step=0.1, style = style,
                          layout = layout)
    αy_val = FloatText(value=alpha_y, description = \
```

```
'Природна народжуваність жертв: $α_y = $', step=0.1, style = style,
                        layout = layout)
 βx_val = FloatText(value=beta_x, description = \
'Природна смертність хижаків: $β_x = $', step=0.1, style = style,
                        layout = layout)
 βy_val = FloatText(value=beta_y, description = \
'"Норма споживаності" жертв: $β_y = $', step=0.1, style = style,
                        layout = layout)
 x0_val = FloatText(value=x0, description = \
'Початкова кількість хижаків: $x_0 = $', step=0.1, style = style,
                        layout = layout)
 y0_val = FloatText(value=y0, description = \
"Початкова кількість жертв: $y_0 = $", step=0.1, style = style,
                        layout = layout)
 models[3] = VBox([description4, interactive(model4, αx=αx_val, αy=αy_val,
                                                βx=βx_val, βy=βy_val,
                                                x0=x0_val, y0=y0_val,
                                                continuous_update=True)])
 display(models[3])
```