

“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ імені І. Сікорського”

Факультет прикладної математики

Кафедра програмного забезпечення комп’ютерних систем

**Домашні завдання**

з дисципліни “Математичне моделювання”

Варіант № 23

|  |  |  |
| --- | --- | --- |
| Виконав  студент V курсу  групи КП-81мп  Сахарчук Тарас Юрійович  (*прізвище, ім’я, по батькові*) |  | Зараховані  “\_\_\_\_” “\_\_\_\_\_\_\_\_\_\_\_\_” 2018 р.  викладачем  Онай Миколою Володимировичем  (*прізвище, ім’я, по батькові*) |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Д.З. | №1 | №2 | №3 | №4 | №5 | №6 | №7 |  |
| Варіант |  |  |  |  |  |  |  |  |
| Оцінка |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  | ∑ = |

Київ 2018

**Домашня робота №1**

1. **Постановка завдання**

Провести моделювання руху каменя шляхом чисельного роз’вязання системи диференціальних рівнянь в будь-якому математичному пакеті. Розглянути випадки коли опір повітря відсутній та з його наявністю. Звіт має містити код програми та графік траєкторії каменя.

1. **Код програми**

p = 23;

v0 = 13 p + 1;

angle = Mod[Ceiling[p^3 / 175 + 9 p] , 90] Degree;

g = 10;

m = 11 p + 1;

A = 0.1;

B = 10^-3;

trajVar = DSolve[{ X''[t] == 0, Y''[t] == -g, X[0] == 5 p, Y[0] == 7 p, X'[0] == v0 Cos[angle], Y'[0] == v0 Sin[angle] }, {X, Y}, t];

Block[

{

flightTime = t /. First[NSolve[{First[Y[t] /. trajVar] == 0, t > 0}, t]],

X = First[X[t] /. trajVar],

Y = First[Y[t] /. trajVar],

ymax := MaxValue[Y, t]

},

Show[

ParametricPlot[{X, Y}, {t, 0, flightTime},

PlotRange -> {{0, X /. t -> Ceiling[flightTime + 1]}, {0, ymax}},

AxesLabel -> {"Distance, m", "High, m"}],

ImageSize -> Full

]

]

trajectoryRes = NDSolve[{  
m X''[t] == -A X'[t] - B (X'[t])^3, m Y''[t] == -m g - A Y'[t] - B (Y'[t])^3,  
X[0] == 5 p, Y[0] == 7 p,  
X'[0] == v0 Cos[angle],  
Y'[0] == v0 Sin[angle],  
WhenEvent[Y[t] == 0, {dragZero = t, "StopIntegration"}]  
}, {X, Y},   {t, 0, 100}];  
Block[{  
 X = First[X[t] /. trajectoryRes],  
 Y = First[Y[t] /. trajectoryRes]  
},  
Show[  
 ParametricPlot[{X, Y}, {t, 0, dragZero}, PlotRange -> {Automatic, {0, Automatic}}, AxesLabel -> {"Dist, meters", "H, meters"} ],  ImageSize -> Full]  
]

1. **Графік польоту**

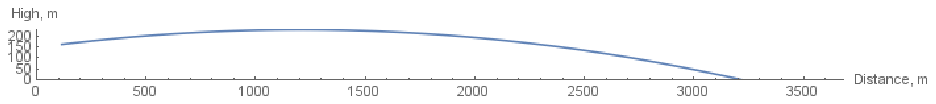


Рис.1. Без врахування опору повітря

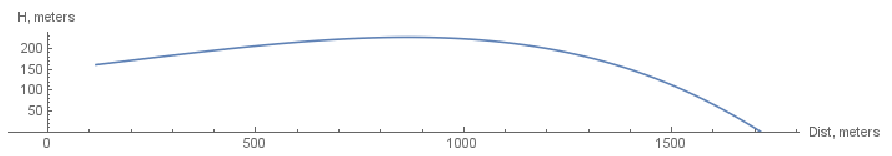


Рис.2. З врахуванням опору повітря

**Домашня робота №2**

1. **Постановка завдання**

Відпускаємо бейсбольний м’ячик з даху, опір повітря пропорційний

швидкості v, вісь Oy направлена вниз. Записати диференціальне рівняння в якому невідомою є швидкість м’яча. Побудувати поле напрямків диференціального рівняння. При заданих початкових умовах в полі напрямків побудувати інтегральні криві та знайти стійкий розв’язок диф. рівняння.

1. **Диференціальне рівняння**
2. **Поле напрямків та інтегральні криві**

Початкова швидкість:

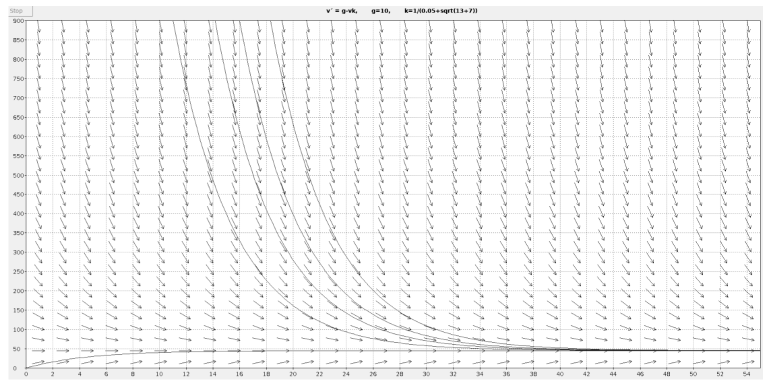


рис. 1. Поле напрямків та інтегральні криві

1. **Стійкий розв’язок**

З побудованого поля напрямків видно, що стійкий розв’язок отримуємо при v(0) ~ 44.5.

**Домашня робота №4**

1. **Постановка завдання**

Під дією сили F=50p+3555=4205 H пружина розтягується на x=p=23м. Початкове положення тіла – x(0)=p/2 =23/2 =11.5 м;

початкова швидкість – v0= x'(0)=−5p= −65 м/с;

маса – кг.

Рівняння руху: x(t)=6,73361cos(36,9679t−6.019)=6,73361cos(36,9679t+0,264182) Параметри:

кутова частота: ω ≈ 36,9679 рад/с

період: T ≈0,169963 с

частота: ν ≈5,88362 Гц

початкова фаза: α ~ 6,019 рад

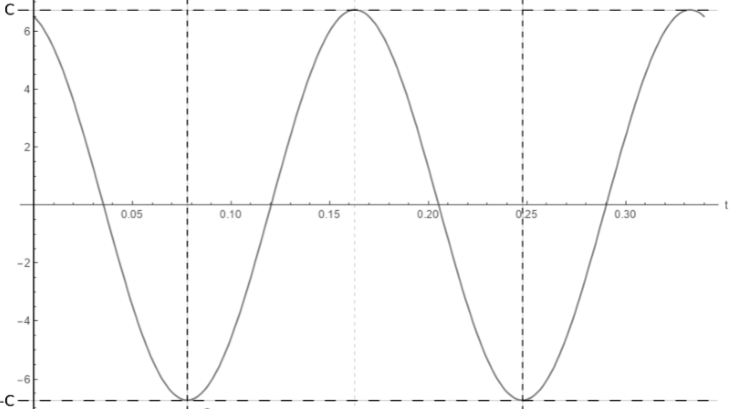
амплітуда: C ≈ 6,73361м

часова затримка: δ ≈ 0,162817 с

1. **Код програми**

p = 13;  
m = (3 p + 1)/p^2;  
x0 = p/2;   
v0 = -5 p;   
f0rce = 50 p + 3555;  
k = f0rce/p;  
w = Sqrt[k/m];   
ODEsolution = X /. First@ DSolve[{X''[t] + w^2 X[t] == 0, X[0] == x0, X'[0] == v0}, X, t];  
{amplitude, alpha} = Block[{   
A = x0,   
B = v0/w,   
C := Sqrt[A^2 + B^2],    
cosAlpha := A/C,   
sinAlpha := B/C,     
tanAlpha := sinAlpha/cosAlpha,     
alphaOffset := \[Piecewise] {{0, A > 0 && B > 0}, {Pi, A < 0}, {2 Pi, A > 0 && B < 0} }},    
{C, ArcTan[tanAlpha] + alphaOffset} ];  
period = 2 Pi/w;  
delay = alpha/w;   
frequency = 1/period;   
Xcos := amplitude Cos[w t \[Dash] alpha]   
Plot[Xcos, {t, 0, 2 period}, ImageSize -> Large,   
GridLines -> {     
{delay - period/2, delay + period/2, {delay, Dashed}},   
{amplitude, -amplitude}   },   
PlotLabel -> Style[Row[{"x(t)=", (Xcos // N)}], Bold],   
AxesLabel -> {"t", "x"}   
]

1. **Графік**



**Домашня робота №7**

1. **Постановка завдання**

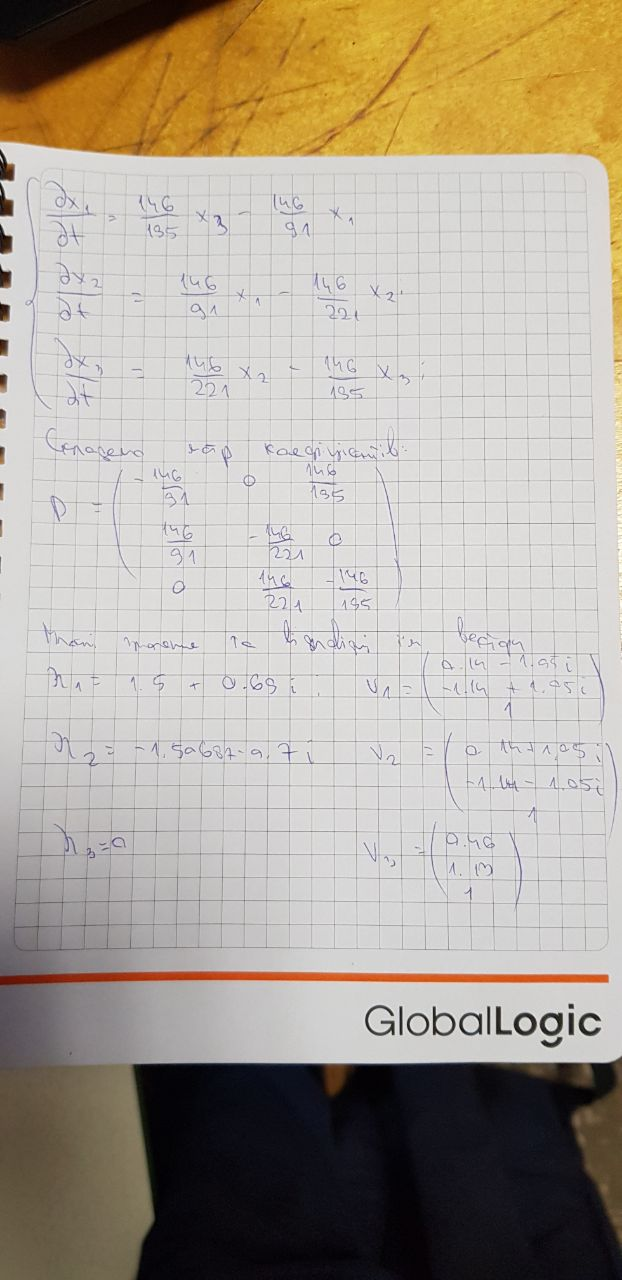
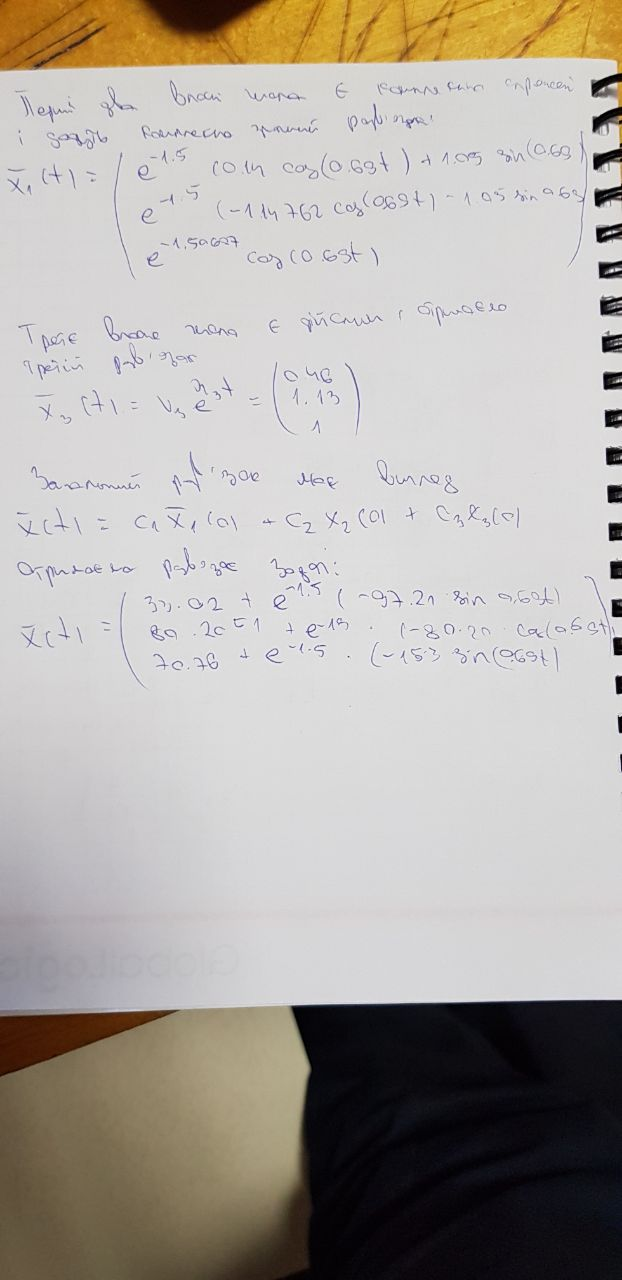
Нехай є замкнена система з трьох баків з об’ємами

V1=91, V2 = 221 та V3 = 195 літрів.

Суміш, що витікає з першого бака, вливається до другого, з другого – до третього, з третього – до першого. Початкова концентрація солі в першому баці x1(0) = 184 кг/літр, в другому та третьому – x2(0)=x3(0)=0. Швидкість потоку рідини між баками – v = 146 л/хв.

Знайти xi(t) та побудувати графіки цих функцій. Розв’язати методом власних значень.

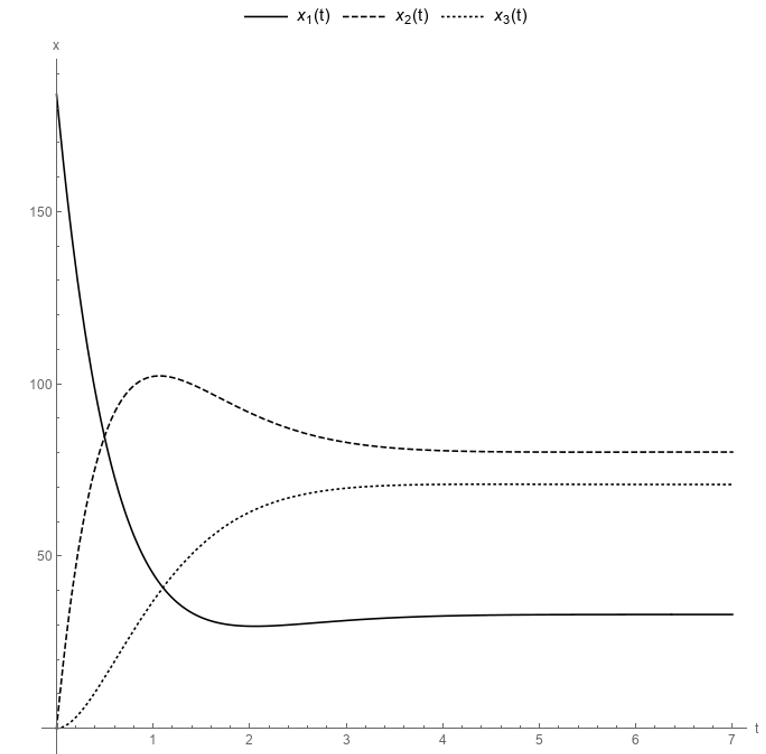
1. **Обчислення**

1. **Код**

varNo = 23;  
V1 = 7 varNo;  
V2 = 17 varNo;  
V3 = 15 varNo;  
x10 = 14 varNo + 2;  
x20 = x30 = 0;  
v = 11 varNo + 3;  
functions = {x1[t], x2[t], x3[t]};   
conditions = {x10, x20, x30};   
P = { {-v/V1, 0, v/V3}, {v/V1, -v/V2, 0}, {0, v/V2, -v/V3} };   
{eigenValues, eigenVectors} = Eigensystem[P];   
x1tComplex = (eigenVectors[[1]] Exp[eigenValues[[1]] t]);   
x1tComplex // N   
x1t = Re /@ x1tComplex // ComplexExpand;   
x2t = Im /@ x1tComplex // ComplexExpand;   
x3t = eigenVectors[[3]] Exp[ eigenValues[[3]] t];   
coefficients = {c1, c2, c3} /. First @ Solve[  ((c1 x1t + c2 x2t + c3 x3t) /. t -> 0 ) == conditions, {c1, c2, c3} ];   
ownSolution = Transpose[{x1t, x2t, x3t}].coefficients;  
referenceSolution = functions /. First @ DSolve[   Join[        
MapThread[#1 == #2 &, {D[functions,t], P.functions}],       
MapThread[#1 == #2 &, {functions /. t -> 0, conditions}]   ],   functions,   t  
]

1. **Побудований графік**



1. **Висновки**

З часом концентрація солі в першому баці спадатиме, у другому – спершу зростатиме, потім почне знижуватися, в третьому – поступово зростатиме.