**상미분 방정식(ODE)**

syms y(t) x

% 1. 방정식: 2y'' + 3y' + y = 5

eq1 = 2\*diff(y, t, 2) + 3\*diff(y, t) + y == 5;

% 2. 방정식: y' = 5 \* sin(t)

eq2 = diff(y, t) == 5 \* sin(t);

% 3. 방정식: x^2 \* y' + exp(x) \* y'' = 2x + y

eq3 = x^2 \* diff(y(x), x) + exp(x) \* diff(y(x), x, 2) == 2\*x + y(x);

% 방정식 출력

disp(eq1)

disp(eq2)

disp(eq3)

* syms y(t) x: y를 시간 t의 함수로, x는 독립 변수로 설정합니다.
* diff(y, t): y의 첫 번째 도함수를 구합니다.
* diff(y, t, 2): y의 두 번째 도함수를 구합니다.
* 각 방정식을 정의한 후, disp 함수를 사용하여 방정식을 출력합니다.

**편미분 방정식(PDE)**

syms v(x, y, z) E(x, y, z) B(x, y, z, t)

% 1. 라플라스 방정식

laplace\_eq = diff(v, x, 2) + diff(v, y, 2) + diff(v, z, 2) == 0;

% 2. 패러데이의 법칙 (맥스웰 방정식)

% E와 B는 벡터 필드이므로, 각각의 컴포넌트에 대해 다룹니다.

syms Ex(x, y, z) Ey(x, y, z) Ez(x, y, z)

syms Bx(x, y, z, t) By(x, y, z, t) Bz(x, y, z, t)

% curl(E) = -dB/dt (패러데이 법칙)

curl\_E = curl([Ex, Ey, Ez], [x, y, z]);

partial\_B\_t = -diff([Bx, By, Bz], t);

% 방정식 출력

disp(laplace\_eq)

disp('Curl of E:');

disp(curl\_E)

disp('Partial derivative of B with respect to t:');

disp(partial\_B\_t)

* **라플라스 방정식**:
* diff(v, x, 2)는 함수 v의 xxx에 대한 2차 편미분을 의미합니다.
* laplace\_eq는 이 편미분들의 합이 0이라는 방정식을 정의합니다.
* **패러데이의 법칙**:
* curl 함수는 벡터 필드 E의 회전을 계산합니다.
* diff([Bx, By, Bz], t)는 벡터 필드 B의 시간에 따른 편미분을 계산합니다.

**1계 상미분 방정식, 2계 편미분 방정식**

syms y(x)

% 1. 1계 상미분 방정식

eq1 = diff(y, x) + 5\*y^2 == -3;

% 2. 2계 편미분 방정식

eq2 = x\*diff(y, x, 2) + y\*diff(y, x) + 5\*y^2 == -3;

% 방정식 출력

disp('1계 상미분 방정식:')

disp(eq1)

disp('2계 편미분 방정식:')

disp(eq2)

* **1계 상미분 방정식**:
* diff(y, x)는 yyy의 xxx에 대한 1차 도함수를 의미합니다.
* eq1는 이 1차 도함수와 y2y^2y2의 결합이 −3-3−3이 되는 방정식을 정의합니다.
* **2계 편미분 방정식**:
* diff(y, x, 2)는 yyy의 xxx에 대한 2차 도함수를 의미합니다.
* eq2는 2차 도함수, 1차 도함수 및 y2y^2y2의 결합이 −3-3−3이 되는 방정식을 정의합니다.

**1계 1차 상미분 방정식, 1계 2차 상미분 방정식, 2계 1차 편미분 방정식, 2계 3차 편미분 방정식**

syms y(x)

% 1. 1계 1차 상미분 방정식

eq1 = diff(y, x) + 5\*y^2 == -3;

% 2. 1계 2차 상미분 방정식

eq2 = (diff(y, x))^2 + 5\*y^2 == -3;

% 3. 2계 1차 편미분 방정식

eq3 = x\*diff(y, x, 2) + y\*diff(y, x) + 5\*y^2 == -3;

% 4. 2계 3차 편미분 방정식

eq4 = x\*(diff(y, x, 2))^3 + y\*diff(y, x) + 5\*y^2 == -3;

% 방정식 출력

disp('1계 1차 상미분 방정식:')

disp(eq1)

disp('1계 2차 상미분 방정식:')

disp(eq2)

disp('2계 1차 편미분 방정식:')

disp(eq3)

disp('2계 3차 편미분 방정식:')

disp(eq4)

* **1계 1차 상미분 방정식**:
* diff(y, x)는 yyy의 xxx에 대한 1차 도함수를 의미합니다.
* eq1은 이 1차 도함수와 y2y^2y2의 결합이 −3-3−3이 되는 방정식을 정의합니다.
* **1계 2차 상미분 방정식**:
* diff(y, x)의 제곱을 사용하여 yyy의 1차 도함수의 제곱과 y2y^2y2의 결합이 −3-3−3이 되는 방정식을 정의합니다.
* **2계 1차 편미분 방정식**:
* diff(y, x, 2)는 yyy의 xxx에 대한 2차 도함수를 의미하며, 이와 yyy의 1차 도함수 및 y2y^2y2의 결합이 −3-3−3이 되는 방정식을 정의합니다.
* **2계 3차 편미분 방정식**:
* diff(y, x, 2)의 3제곱을 사용하여, 이와 yyy의 1차 도함수 및 y2y^2y2의 결합이 −3-3−3이 되는 방정식을 정의합니다.

**1계 1차 선형 상미분 방정식, 1계 2차 비선형 상미분 방정식**

syms y(x)

% 1. 1계 1차 선형 상미분 방정식

eq1 = diff(y, x) + 5\*y == -3;

% 2. 1계 2차 비선형 상미분 방정식

eq2 = (diff(y, x))^2 + 5\*y^2 == -3;

% 방정식 출력

disp('1계 1차 선형 상미분 방정식:')

disp(eq1)

disp('1계 2차 비선형 상미분 방정식:')

disp(eq2)

* **1계 1차 선형 상미분 방정식**:
* diff(y, x)는 yyy의 xxx에 대한 1차 도함수를 의미합니다.
* eq1은 이 1차 도함수와 yyy가 선형적으로 결합된 방정식을 정의합니다.
* **1계 2차 비선형 상미분 방정식**:
* diff(y, x)의 제곱을 사용하여 yyy의 1차 도함수의 제곱과 y2y^2y2의 결합이 비선형적으로 나타나는 방정식을 정의합니다.

**1계 1차 비선형 상미분 방정식**

syms y(x)

% 1계 1차 비선형 상미분 방정식

eq = diff(y, x) + sin(y) == -3;

% 방정식 출력

disp('1계 1차 비선형 상미분 방정식:')

disp(eq)

* diff(y, x)는 yyy의 xxx에 대한 1차 도함수를 의미합니다.
* sin(y)는 yyy의 사인 함수입니다.
* eq는 이 두 항을 결합하여 비선형 미분 방정식을 정의합니다.

**음함수형 (Implicit Form), 양함수형 (Explicit Form)**

syms x y(x) y\_prime y\_double\_prime F

% 음함수형 (Implicit Form)

% 예를 들어, F(x, y, y', y'') = 0로 정의할 수 있습니다.

F = x + y + diff(y, x) + diff(y, x, 2); % 예시 함수

implicit\_eq = F == 0;

% 양함수형 (Explicit Form)

% 예를 들어, y'' = f(x, y, y')로 표현할 수 있습니다.

explicit\_eq = diff(y, x, 2) == x + y + diff(y, x); % 예시 함수

% 방정식 출력

disp('음함수형 (Implicit Form):')

disp(implicit\_eq)

disp('양함수형 (Explicit Form):')

disp(explicit\_eq)

**초깃값 문제**

% Define the differential equation dy/dx = f(x, y)

syms y(x)

ode = diff(y, x) == 1; % dy/dx = 1

% Define the direction field grid

[xVals, yVals] = meshgrid(-5:0.5:5, -5:0.5:5);

dy = ones(size(xVals)); % Since dy/dx = 1, dx/dx = 1

dx = dy; % This is because the slope field lines are straight

% Plot the direction field (left graph)

figure;

quiver(xVals, yVals, dx, dy, 'k'); % Draw direction field lines

hold on;

% Solve the differential equation with initial condition y(0) = -3

cond = y(0) == -3;

ySol(x) = dsolve(ode, cond);

% Plot the specific solution (right graph)

fplot(ySol, [-5, 5], 'r', 'LineWidth', 2);

plot(0, -3, 'ro', 'MarkerSize', 8, 'MarkerFaceColor', 'r'); % Initial condition point

% Labels and grid

xlabel('x');

ylabel('y');

title('Direction Field and Solution with Initial Condition');

grid on;

% Set axis limits

xlim([-5 5]);

ylim([-5 5]);

% Add annotations for initial condition

text(0.2, -3.5, 'y(0) = -3', 'FontSize', 12, 'Color', 'k');

hold off;

* **방향 필드**:
* quiver 함수를 사용하여 방향 필드를 그립니다. 이 필드는 주어진 미분 방정식 dydx=1\frac{dy}{dx} = 1dxdy�=1의 각 점에서의 기울기를 나타냅니다.
* **특정 해와 초기 조건**:
* dsolve를 사용하여 미분 방정식을 초기 조건 y(0)=−3y(0) = -3y(0)=−3으로 풀어 해를 구합니다.
* fplot을 사용하여 이 해를 그래프에 빨간색 선으로 표시합니다.
* **그래프 설정**:
* plot을 사용하여 초기 조건 점 (0,−3)(0, -3)(0,−3)을 강조 표시합니다.
* 그래프에 제목, 축 레이블, 초기 조건에 대한 텍스트 주석을 추가합니다.