

問 1 (4)

```
In [1]: using SymPy

# x を記号変数として宣言する。
# ここでは x が独立変数である。
@syms x

# y を未知関数として宣言する。
# 数学でいう y(x) を扱うための準備である。
y = SymFunction("y")

# 微分方程式
#
#  $y'' + 2y' - 3y = 0$ 
#
# を SymPy の等式として定義する。
#
#  $\text{diff}(y(x), x)$  は  $y'(x)$  を表す。
#  $\text{diff}(y(x), x, 2)$  は  $y''(x)$  を表す。
eq = Eq(
    diff(y(x), x, 2) + 2 * diff(y(x), x) - 3 * y(x),
    0
)

# 定義した微分方程式を表示する。
display(eq)

# 微分方程式 eq を、未知関数 y(x) について解く。
sol = dsolve(eq, y(x))

# 解を表示する。
#
# 結果は例えば
#
#  $\text{Eq}(y(x), C1 \cdot \exp(-3 \cdot x) + C2 \cdot \exp(x))$ 
#
# のような形で表示される。
display(sol)

# sol は
#
#  $y(x) = \dots$ 
#
# という等式である。
# rhs(sol) により、その右辺だけを取り出す。
sol_rhs = rhs(sol)

# 得られた解が本当に微分方程式を満たすか確認する。
#
# 元の微分方程式は
#
#  $y'' + 2y' - 3y = 0$ 
#
# なので、
#
```

```

# y'' + 2y' - 3y
#
# を計算して 0 になるかを調べる。
check = simplify(
    diff(sol_rhs, x, 2) + 2 * diff(sol_rhs, x) - 3 * sol_rhs
)

# check が 0 ならば、得られた解は確かに微分方程式を満たしている。
display(check)

```

$$-3y(x) + 2\frac{d}{dx}y(x) + \frac{d^2}{dx^2}y(x) = 0$$

$$y(x) = C_1e^{-3x} + C_2e^x$$

0

問 2 (3)

In [2]: **using** SymPy

```

# x を記号変数として宣言する。
# ここでは x が独立変数である。
@syms x

# y を未知関数として宣言する。
# 数学でいう y(x) を扱うための準備である。
y = SymFunction("y")

# D = d/dx とすると、
#
# (D - 1)(D - 2)
#
# は
#
# D^2 - 3D + 2
#
# である。
#
# したがって
#
# (D - 1)(D - 2)y = x e^x
#
# は
#
# y'' - 3y' + 2y = x e^x
#
# という微分方程式である。
eq = Eq(
    diff(y(x), x, 2) - 3 * diff(y(x), x) + 2 * y(x),
    x * exp(x)
)

# 微分方程式を表示する。
display(eq)

# 微分方程式を y(x) について解く。
sol = dsolve(eq, y(x))

```

```

# 解を表示する。
#
# dsolve は一般解を返すので、
# 同次方程式の解 C1*exp(x), C2*exp(2*x) も含まれる。
display(sol)

# sol の右辺を取り出す。
sol_rhs = rhs(sol)

# 今回ほしいのは
#
#      1 / ((D - 1)(D - 2)) x e^x
#
# であり、これは「特解」を求める操作である。
#
# そのため、一般解に含まれる任意定数 C1, C2 を 0 にして、
# 特解だけを取り出す。
#
# SymPy が使う積分定数 C1, C2 を記号として用意する。
@syms C1 C2

particular = simplify(
    sol_rhs.subs(C1, 0).subs(C2, 0)
)

# 特解を表示する。
display(particular)

# 得られた particular が本当に
#
#      (D - 1)(D - 2)y = x e^x
#
# を満たすか確認する。
#
# つまり
#
#      y'' - 3y' + 2y - x e^x
#
# を計算して 0 になるか調べる。
check = simplify(
    diff(particular, x, 2)
    - 3 * diff(particular, x)
    + 2 * particular
    - x * exp(x)
)

# check が 0 ならば、特解は正しい。
display(check)

```

$$2y(x) - 3\frac{d}{dx}y(x) + \frac{d^2}{dx^2}y(x) = xe^x$$

$$y(x) = \left(C_1 + C_2 e^x - \frac{x^2}{2} - x \right) e^x$$

$$\frac{x(-x-2)e^x}{2}$$

問 3 (1)

```
In [3]: using SymPy

# x を記号変数として宣言する。
# ここでは x が独立変数である。
@syms x

# y を未知関数として宣言する。
# 数学でいう y(x) を扱うための準備である。
y = SymFunction("y")

# 問題は
#
#  $D(D+3)y = e^{(3x)} + x$ 
#
# である。
#
# ここで  $D = d/dx$  なので、
#
#  $D(D+3)y = D^2 y + 3D y$ 
#
# である。
#
# したがって微分方程式は
#
#  $y'' + 3y' = e^{(3x)} + x$ 
#
# となる。
eq = Eq(
    diff(y(x), x, 2) + 3 * diff(y(x), x),
    exp(3 * x) + x
)

# 定義した微分方程式を表示する。
display(eq)

# 微分方程式 eq を、未知関数 y(x) について解く。
sol = dsolve(eq, y(x))

# 解を表示する。
#
# 結果は例えば
#
#  $Eq(y(x), C1 + C2*exp(-3*x) + exp(3*x)/18 + x^2/6 - x/9)$ 
#
# のような形で表示される。
display(sol)

# sol は
#
#  $y(x) = \dots$ 
#
# という等式である。
# rhs(sol) により、その右辺だけを取り出す。
sol_rhs = rhs(sol)

# 得られた解が本当に微分方程式を満たすか確認する。
```

```

#
# 元の微分方程式は
#
#   y'' + 3y' = e^(3x) + x
#
#   なので、
#
#   y'' + 3y' - e^(3x) - x
#
#   を計算して 0 になるかを調べる。
check = simplify(
    diff(sol_rhs, x, 2)
    + 3 * diff(sol_rhs, x)
    - exp(3 * x)
    - x
)

# check が 0 ならば、得られた解は確かに微分方程式を満たしている。
display(check)

```

$$3 \frac{d}{dx} y(x) + \frac{d^2}{dx^2} y(x) = x + e^{3x}$$

$$y(x) = C_1 + C_2 e^{-3x} + \frac{x^2}{6} - \frac{x}{9} + \frac{e^{3x}}{18}$$

0

問 4 (1)

In [4]: **using** SymPy

```

# x を記号変数として宣言する。
#   ここでは x が独立変数である。
@syms x

# y を未知関数として宣言する。
#   数学でいう y(x) を扱うための準備である。
y = SymFunction("y")

#   問題は
#
#   (D - 2)(D + 3)y = e^(3x)
#
#   である。
#
#   ここで D = d/dx なので、
#
#   (D - 2)(D + 3)
#   = D^2 + D - 6
#
#   である。
#
#   したがって微分方程式は
#
#   y'' + y' - 6y = e^(3x)
#
#   となる。
eq = Eq(

```

```

diff(y(x), x, 2) + diff(y(x), x) - 6 * y(x),
exp(3 * x)
)

# 微分方程式を表示する。
display(eq)

# 初期条件
#
#    $y(0) = 1$ 
#    $y'(0) = 0$ 
#
# を指定する。
#
#  $y(0)$  は  $y(x)$  に  $x=0$  を代入したもの。
#  $\text{diff}(y(x), x).\text{subs}(x, 0)$  は  $y'(x)$  に  $x=0$  を代入したもの。
ics = Dict(
    y(0) => 1,
    diff(y(x), x).subs(x, 0) => 0
)

# 初期条件つきで微分方程式を解く。
sol = dsolve(eq, y(x), ics=ics)

# 解を表示する。
#
# 結果は例えば
#
#    $\text{Eq}(y(x), \exp(3*x)/6 + 2*\exp(2*x)/5 + 13*\exp(-3*x)/30)$ 
#
# のような形で表示される。
display(sol)

# sol は
#
#    $y(x) = \dots$ 
#
# という等式である。
# rhs(sol) により、その右辺だけを取り出す。
sol_rhs = rhs(sol)

# 得られた解が微分方程式を満たすか確認する。
#
# 元の微分方程式は
#
#    $y'' + y' - 6y = e^{(3x)}$ 
#
# なので、
#
#    $y'' + y' - 6y - e^{(3x)}$ 
#
# を計算して 0 になるかを調べる。
check_eq = simplify(
    diff(sol_rhs, x, 2)
    + diff(sol_rhs, x)
    - 6 * sol_rhs
    - exp(3 * x)
)

# 得られた解が初期条件  $y(0)=1$  を満たすか確認する。

```

```

check_ic1 = simplify(
    sol_rhs.subs(x, 0) - 1
)

# 得られた解が初期条件 y'(0)=0 を満たすか確認する。
check_ic2 = simplify(
    diff(sol_rhs, x).subs(x, 0) - 0
)

# すべて 0 になれば、
# 微分方程式と初期条件をすべて満たしている。
display(check_eq)
display(check_ic1)
display(check_ic2)

```

$$-6y(x) + \frac{d}{dx}y(x) + \frac{d^2}{dx^2}y(x) = e^{3x}$$

$$y(x) = \frac{e^{3x}}{6} + \frac{2e^{2x}}{5} + \frac{13e^{-3x}}{30}$$

0

0

0

In []: