

# Elementary maths

This is a collection of basic mathematical computations using `sympy`. The main purpose is to demonstrate the use of `\py` and `\py*`. Note that `sympy 1.1.1` appears unable to simplify  $\tanh(\log(x))$  (compare `rhs.108` shown below against `ans.108` shown in the [Mathematica](#) examples). Note also the separate computations for the left and right hand sides of results 108, 109 and 110.

```
from sympy import *
x, y, z, a, b, c = symbols('x y z a b c')
ans = expand((a+b)**3)
ans = factor(-2*x+2*x+a*x-x**2+a*x**2-x**3)
ans = solve(x**2-4, x)
ans = solve([2*a-b - 3, a+b+c - 1, -b+c - 6], [a,b,c])
ans = N(pi,50)
ans = apart(1/((1 + x)*(5 + x)))
ans = together((1/(1 + x) - 1/(5 + x))/4)
ans = simplify(tanh(log(x)))
ans = simplify(tanh(I*x))
ans = simplify(sinh(3*x) - 3*sinh(x) - 4*(sinh(x))**3)
ans = tanh(log(x))
ans = tanh(UnevaluatedExpr(I*x))
ans = sinh(3*x) - 3*sinh(x) - 4*(sinh(x))**3
```

```
# py (ans.101,ans)
# py (ans.102,ans)
# py (ans.103,ans)
# py (ans.104,ans)
# py (ans.105,ans)
# py (ans.106,ans)
# py (ans.107,ans)
# py (rhs.108,ans)
# py (rhs.109,ans)
# py (rhs.110,ans)
# py (lhs.108,ans)
# py (lhs.109,ans)
# py (lhs.110,ans)
```

```
\begin{align*}
&\&\py*{ans.101}\\
&\&\py*{ans.102}\\
&\&\py*{ans.103}\\
&\&\py*{ans.104}\\
&\&\py*{ans.105}\\
&\&\py*{ans.106}\\
&\&\py*{ans.107}\\
\py{lhs.108} &= \Py{rhs.108}\\
\py{lhs.109} &= \Py{rhs.109}\\
\py{lhs.110} &= \Py{rhs.110}
\end{align*}
```

$$\text{ans.101} := a^3 + 3a^2b + 3ab^2 + b^3$$

$$\text{ans.102} := -x(-a+x)(x+1)$$

$$\text{ans.103} := [-2, 2]$$

$$\text{ans.104} := \left\{ a : \frac{1}{5}, b : -\frac{13}{5}, c : \frac{17}{5} \right\}$$

$$\text{ans.105} := 3.1415926535897932384626433832795028841971693993751$$

$$\text{ans.106} := -\frac{1}{4(x+5)} + \frac{1}{4(x+1)}$$

$$\text{ans.107} := \frac{1}{(x+1)(x+5)}$$

$$\tanh(\log(x)) = \tanh(\log(x)) \quad (\text{rhs.108})$$

$$\tanh(ix) = i \tan(x) \quad (\text{rhs.109})$$

$$-4 \sinh^3(x) - 3 \sinh(x) + \sinh(3x) = 0 \quad (\text{rhs.110})$$

# Linear Algebra

```

from sympy import linsolve
lamda = Symbol('lamda')
mat = Matrix([[2,3], [5,4]])
eig1 = mat.eigenvects()[0][0]
eig2 = mat.eigenvects()[1][0]
v1 = mat.eigenvects()[0][2][0]
v2 = mat.eigenvects()[1][2][0]
eig = simplify(Matrix([eig1,eig2]))
vec = simplify(5*Matrix([]).col_insert(0,v1)
               .col_insert(1,v2))
det = expand((mat - lamda * eye(2)).det())
rhs = Matrix([[3], [7]])
ans = list(linsolve((mat,rhs),x,y))[0]

```

*# py (ans.201,mat)*  
*# 1st eigenvalue*  
*# 2nd eigenvalue*  
*# 1st eigenvector*  
*# 2nd eigenvector*  
*# py (ans.202,eig)*  
*# py (ans.203,vec)*  
*# py (ans.204,det)*  
*# py (ans.205,rhs)*  
*# py (ans.206,ans)*

```

\begin{align*}
&\&\py*{ans.201}\\
&\&\py*{ans.202}\\
&\&\py*{ans.203}\\
&\&\py*{ans.204}\\
&\&\py*{ans.205}\\
&\&\py*{ans.206}
\end{align*}

```

$$\begin{aligned}
\text{ans.201} &:= \begin{bmatrix} 2 & 3 \\ 5 & 4 \end{bmatrix} \\
\text{ans.202} &:= \begin{bmatrix} -1 \\ 7 \end{bmatrix} \\
\text{ans.203} &:= \begin{bmatrix} -5 & 3 \\ 5 & 5 \end{bmatrix} \\
\text{ans.204} &:= \lambda^2 - 6\lambda - 7 \\
\text{ans.205} &:= \begin{bmatrix} 3 \\ 7 \end{bmatrix} \\
\text{ans.206} &:= \left( \frac{9}{7}, \frac{1}{7} \right)
\end{aligned}$$

# Limits

```
n, dx = symbols('n dx')
ans = limit(sin(4*x)/x,x,0)           # py (ans.301,ans)
ans = limit(2**x/x,x,oo)              # py (ans.302,ans)
ans = limit(((x+dx)**2 - x**2)/dx, dx,0) # py (ans.303,ans)
ans = limit((4*n + 1)/(3*n - 1),n,oo)  # py (ans.304,ans)
ans = limit((1+(a/n))**n,n,oo)        # py (ans.305,ans)
```

```
\begin{align*}
&\&\texttt{py}\{ans.301\}\\
&\&\texttt{py}\{ans.302\}\\
&\&\texttt{py}\{ans.303\}\\
&\&\texttt{py}\{ans.304\}\\
&\&\texttt{py}\{ans.305\}
\end{align*}
```

```
ans.301 := 4
ans.302 := ∞
ans.303 := 2x
ans.304 :=  $\frac{4}{3}$ 
ans.305 :=  $e^a$ 
```

# Series

```
ans = series((1 + x)**(-2), x, 1, 6)   # py (ans.401,ans)
ans = series(exp(x), x, 0, 6)          # py (ans.402,ans)
ans = Sum(1/n**2, (n,1,50)).doit()     # py (ans.403,ans)
ans = Sum(1/n**4, (n,1,oo)).doit()     # py (ans.404,ans)
```

```
\begin{align*}
&\&\texttt{py}\{ans.401\}\\
&\&\texttt{py}\{ans.402\}\\
&\&\texttt{py}\{ans.403\}\\
&\&\texttt{py}\{ans.404\}
\end{align*}
```

```
ans.401 :=  $\frac{1}{2} + \frac{3}{16} (x-1)^2 - \frac{1}{8} (x-1)^3 + \frac{5}{64} (x-1)^4 - \frac{3}{64} (x-1)^5 - \frac{x}{4} + O((x-1)^6; x \rightarrow 1)$ 
ans.402 :=  $1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24} + \frac{x^5}{120} + O(x^6)$ 
ans.403 :=  $\frac{3121579929551692678469635660835626209661709}{1920815367859463099600511526151929560192000}$ 
ans.404 :=  $\frac{\pi^4}{90}$ 
```

# Calculus

This example shows how \Py can be used to set the equation tag on the far right hand side.

```
ans = diff(x*sin(x),x) # py (ans.501,ans)
ans = diff(x*sin(x),x).subs(x,pi/4) # py (ans.502,ans)
ans = integrate(2*sin(x)**2, (x,a,b)) # py (ans.503,ans)
ans = Integral(2*exp(-x**2), (x,0,oo)) # py (lhs.504,ans)
ans = ans.doit() # py (ans.504,ans)
ans = Integral(Integral(x**2 + y**2, (y,0,x)), (x,0,1)) # py (lhs.505,ans)
ans = ans.doit() # py (ans.505,ans)
```

```
\begin{align*}
&\&\py*{ans.501}\\
&\&\py*{ans.502}\\
&\&\py*{ans.503}\\
&\py{lhs.504}\&=\Py{ans.504}\\
&\py{lhs.505}\&=\Py{ans.505}
\end{align*}
```

$$\text{ans.501} := x \cos(x) + \sin(x)$$

$$\text{ans.502} := \frac{\pi}{8}\sqrt{2} + \frac{\sqrt{2}}{2}$$

$$\text{ans.503} := -a + b + \sin(a) \cos(a) - \sin(b) \cos(b)$$

$$\int_0^{\infty} 2e^{-x^2} dx = \sqrt{\pi} \quad (\text{ans.504})$$

$$\int_0^1 \int_0^x (x^2 + y^2) dy dx = \frac{1}{3} \quad (\text{ans.505})$$

# Differential equations

```

y = Function('y')
C1, C2 = symbols('C1 C2')

ode = Eq(y(x).diff(x) + y(x), 2*a*sin(x))
sol = expand(dsolve(ode,y(x)).rhs) # py (ans.601,sol)
cst = solve([sol.subs(x,0)],dict=True)
sol = sol.subs(cst[0]) # py (ans.602,sol)

ode = Eq(y(x).diff(x,2) + y(x), 0)
sol = expand(dsolve(ode,y(x)).rhs) # py (ans.603,sol)
cst = solve([sol.subs(x,0),sol.diff(x).subs(x,0)-1],dict=True)
sol = sol.subs(cst[0]) # py (ans.604,sol)

ode = Eq(y(x).diff(x,2) + 5*y(x).diff(x) - 6*y(x), 0)
sol = expand(dsolve(ode,y(x)).rhs) # py (ans.605,sol)
sol = sol.subs({C1:2,C2:3}) # py (ans.606,sol)

```

```

\begin{align*}
&\&\text{\py*{ans.601}}\\
&\&\text{\py*{ans.602}}\\
&\&\text{\py*{ans.603}}\\
&\&\text{\py*{ans.604}}\\
&\&\text{\py*{ans.605}}\\
&\&\text{\py*{ans.606}}
\end{align*}

```

$$\text{ans.601} := C_1 e^{-x} + a \sin(x) - a \cos(x)$$

$$\text{ans.602} := a \sin(x) - a \cos(x) + a e^{-x}$$

$$\text{ans.603} := C_1 \sin(x) + C_2 \cos(x)$$

$$\text{ans.604} := \sin(x)$$

$$\text{ans.605} := C_1 e^{-6x} + C_2 e^x$$

$$\text{ans.606} := 3e^x + 2e^{-6x}$$